

日 本 国 特 許 庁
PATENT OFFICE
JAPANESE GOVERNMENT



別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日

Date of Application:

2000年 2月15日

出 願 番 号

Application Number:

特願2000-036314

出 願 人

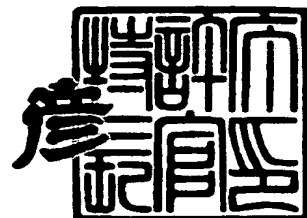
Applicant(s):

インターナショナル・ビジネス・マシーンズ・コーポレイション

2000年 3月31日

特許庁長官
Commissioner,
Patent Office

近 藤 隆 彦



出証番号 出証特2000-3021642

【書類名】 特許願

【整理番号】 JP9000034

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 13/00

【発明者】

【住所又は居所】 神奈川県大和市下鶴間 1 6 2 3 番地 1 4 日本アイ・ピー・エム株式会社 大和事業所内

【氏名】 大門 昭

【発明者】

【住所又は居所】 神奈川県大和市下鶴間 1 6 2 3 番地 1 4 日本アイ・ピー・エム株式会社 大和事業所内

【氏名】 吉田 洋一

【発明者】

【住所又は居所】 神奈川県大和市下鶴間 1 6 2 3 番地 1 4 日本アイ・ピー・エム株式会社 大和事業所内

【氏名】 金武 貢

【特許出願人】

【識別番号】 390009531

【住所又は居所】 アメリカ合衆国 1 0 5 0 4、ニューヨーク州アーモンク (番地なし)

【氏名又は名称】 インターナショナル・ビジネス・マシーンズ・コーポレーション

【代理人】

【識別番号】 100086243

【弁理士】

【氏名又は名称】 坂口 博

【選任した代理人】

【識別番号】 100091568

【弁理士】

【氏名又は名称】 市位 嘉宏

【手数料の表示】

【予納台帳番号】 024154

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9706050

【包括委任状番号】 9704733

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 コンテンツ情報の取得方法、コンテンツ情報を取得するためのソフトウェア・プロダクト、コラボレーション・システム及びコラボレーション・サーバー

【特許請求の範囲】

【請求項 1】

コンテンツ情報を表示するカスタマー側ブラウザを搭載するカスタマー側情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報をエージェント側情報端末が取得するコンテンツ情報の取得方法であって、

(a) 前記エージェント側情報端末がコラボレーション・サーバーと通信接続する段階と、

(b) 前記エージェント側情報端末を一意に特定する U A I を生成する段階と、

(c) 前記コラボレーション・サーバーから、個人情報を入力し、前記エージェント側情報端末から前記コラボレーション・サーバーへ送信するためのエージェント側制御ウィンドウを生成するエージェント側アプレットを取得する段階と、

(d) 前記エージェント側アプレットを起動し、前記エージェント側制御ウィンドウを生成する段階と、

(e) エージェントのログオン指示操作に応答して、エージェントのユーザー ID を含むログオン要求を前記エージェント側情報端末から前記コラボレーション・サーバーに送信する段階と、

(f) 前記カスタマー側情報端末が前記コラボレーション・サーバーと通信接続する段階と、

(g) 前記カスタマー側情報端末を一意に特定する U A I を生成する段階と、

(h) 前記コラボレーション・サーバーから、個人情報を入力し、前記カスタマー側情報端末から前記コラボレーション・サーバーへ送信するためのカスタマー側制御ウィンドウを生成するカスタマー側第 1 アプレットを取得する段階と、

(i) 前記カスタマー側第 1 アプレットを起動し、前記カスタマー側制御ウイ

ンドウを生成する段階と、

(j) カスタマーのログオン指示操作に応答して、カスタマーの個人情報および問い合わせ種別情報を含むログオン要求を前記カスタマー側情報端末から前記コラボレーション・サーバーに送信する段階と、

(k) 前記問い合わせ種別情報に対応したカスタマー待ち行列に該カスタマーのレコードを追加する段階と、

(l) 前記対応したカスタマー待ち行列における前記カスタマーのレコードの待ち状態を検査する段階と、

(m) 該検査結果をカスタマー側情報端末に通知する段階と、

(n) 前記問い合わせ種別情報に対応したエージェントを前記カスタマーにアサインする段階と、

(o) 前記エージェント側情報端末にセッションの開始を通知する段階と、

(p) 前記カスタマー側情報端末にセッションの開始を通知する段階と、

(q) 前記カスタマー側情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーからコンテンツ情報を取得するに際し、前記コラボレーション・サーバーが、前記コンテンツ情報に前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出するカスタマー側第2アプレットを埋めこむ段階と、

(r) 変更後のコンテンツ情報を含むページのURL、前記変更後のコンテンツ情報を含むページの親のページのURL、前記親のページが持つ子のページの数、その親のページに対する前記変更後のコンテンツ情報を含むページの位置を含む変更後ページ情報を取得するカスタマー側スクリプトを埋めこむ段階と、

(s) 前記変化後ページ情報を前記コラボレーション・サーバに送信するとともに、エージェント側から送信された変化後ページ情報を取得したことに応答して、該送信された変化後ページ情報が現在カスタマー側ブラウザで表示されているページに該当するか否かを検査し、該当しない場合には、該送信された変化後ページ情報に対応したページを表示するように前記カスタマー側ブラウザに指示するツリー・マネージャーを埋めこむ段階と、

(t) 前記カスタマー側第2アプレットが前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出したことに応答して、変更後ページ情報を前記コ

ラボレーション・サーバーに送信する段階と、

(u) 前記コラボレーション・サーバーが前記エージェント側情報端末に変更後ページ情報を送信する段階と、

(v) 前記エージェント側情報端末において、現在エージェント側ブラウザが表示中のページが変更後ページ情報に対応するか否か判断する段階と、

(w) 前記エージェント側ブラウザが現在表示中のページが変更後ページ情報に対応しないと判断された場合、前記エージェント側ブラウザに変更後ページ情報に対応するページをロードさせる段階と、

を含むコンテンツ情報の取得方法。

【請求項 2】

コンテンツ情報を表示する第 1 のブラウザを搭載する第 1 の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第 2 の情報端末が取得するコンテンツ情報の取得方法であって、

(a) 前記第 1 の情報端末とコラボレーション・サーバーの間を通信接続する段階と、

(b) 前記第 2 の情報端末と前記コラボレーション・サーバーの間を通信接続する段階と、

(c) 前記第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋めこむ段階と、

(c) 前記第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信する段階と、

(e) 前記コラボレーション・サーバーが前記第 2 の情報端末に前記変更後のコンテンツ特定情報を送信する段階と、

(f) 前記第 2 の情報端末において、前記変更後のコンテンツ情報を取得する段階と、

を含むコンテンツ情報の取得方法。

【請求項 3】

コンテンツ情報を表示する第 1 のブラウザを搭載する第 1 の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第 2 の情報端末が取得するコンテンツ情報の取得方法であって、

(a) コラボレーション・サーバーに接続する第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋め込んで前記第 1 の情報端末に送信する段階と、

(b) 前記第 1 の情報端末の前記クライアント・コントローラーが送信した変更後のコンテンツ特定情報を前記コラボレーション・サーバーが受信する段階と、

(c) 前記コラボレーション・サーバーに通信接続された第 2 の情報端末において、前記変更後のコンテンツ情報を取得することを可能にするために、前記変更後のコンテンツ特定情報を前記第 2 の情報端末に送信する段階と、

を含むコンテンツ情報の取得方法。

【請求項 4】

コンテンツ情報を表示する第 1 のブラウザを搭載する第 1 の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第 2 の情報端末が取得することをサポートするソフトウェア・プロダクトであって、

(a) コラボレーション・サーバーに接続する第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋め込むことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(b) 前記第 1 の情報端末から送信された変更後のコンテンツ特定情報を前記第 2 の情報端末に転送することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

を含むソフトウェアプロダクト。

【請求項 5】

コンテンツ情報を表示するカスタマー側ブラウザを搭載するカスタマー側情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報をエージェント側情報端末が取得することをサポートするソフトウェア・プロダクトであって、

(a) コラボレーション・サーバーと通信接続するエージェント側情報端末を一意に特定する U A I を生成することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(b) 前記エージェント側情報端末において、個人情報を入力し、前記エージェント側情報端末から前記コラボレーション・サーバーへ送信するためのエージェント側制御ウインドウを生成し、エージェントのログオン指示操作に応答して、エージェントのユーザー I D を含むログオン要求を前記エージェント側情報端末から前記コラボレーション・サーバーに送信するエージェント側アプリレットと、

(c) 前記コラボレーション・サーバーと通信接続するカスタマー側情報端末を一意に特定する U A I を生成することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(d) 前記カスタマー側情報端末において、個人情報を入力し、前記カスタマー側情報端末から前記コラボレーション・サーバーへ送信するためのカスタマー側制御ウインドウを生成し、カスタマーのログオン指示操作に応答して、カスタマーの個人情報および問い合わせ種別情報を含むログオン要求を前記カスタマー側情報端末から前記コラボレーション・サーバーに送信するカスタマー側第 1 アプリレットと、

(e) 前記問い合わせ種別情報に対応したエージェントを前記カスタマーにアサインすることを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(f) 前記エージェント側情報端末にセッションの開始を通知することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(g) 前記カスタマー側情報端末にセッションの開始を通知することを前記コ

ラボレーション・サーバーに指示するためのプログラムコードと、

(h) 前記カスタマー側情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーからコンテンツ情報を取得するに際し、前記コラボレーション・サーバーが、前記コンテンツ情報に前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出するカスタマー側第2アプレットを埋めこむことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(i) 変更後のコンテンツ情報を含むページのURL、前記変更後のコンテンツ情報を含むページの親のページのURL、前記親のページが持つ子のページの数、その親のページに対する前記変更後のコンテンツ情報を含むページの位置を含む変更後ページ情報を取得するカスタマー側スクリプトを埋めこむことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(j) 前記変化後ページ情報を前記コラボレーション・サーバに送信するとともに、エージェント側から送信された変化後ページ情報を取得したことに応答して、該送信された変化後ページ情報が現在カスタマー側ブラウザで表示されているページに該当するか否かを検査し、該当しない場合には、該送信された変化後ページ情報に対応したページを表示するように前記カスタマー側ブラウザに指示するツリー・マネージャーを埋めこむことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(k) 前記カスタマー側第2アプレットが前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出したことに応答して、前記カスタマー側情報端末から送信された変更後ページ情報を前記エージェント側情報端末に送信することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

を含むソフトウェアプロダクト。

【請求項6】

コンテンツ情報を表示するカスタマー側ブラウザを搭載するカスタマー側情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報をエージェント側情報端末が取得することをサポートするコラボレーション・システムであって、

(a) カスタマー側ブラウザを搭載するカスタマー側情報端末と、

- (b) エージェント側ブラウザを搭載するエージェント側情報端末と、
- (c) 前記エージェント側情報端末前記カスタマー側情報端末との間の通信接続をサポートする HTTPブリッジと、
- (d) 前記エージェント側ブラウザ及び前記カスタマー側ブラウザを夫々一意に特定する U A I を生成する段階と、
- (e) 前記コラボレーション・サーバーから、個人情報を入力し、前記エージェント側情報端末から前記コラボレーション・サーバーへ送信するためのエージェント側制御ウインドウを生成し、エージェントのログオン指示操作に応答して、エージェントのユーザー I D を含むログオン要求を前記エージェント側情報端末から前記コラボレーション・サーバーに送信するエージェント側アプレットと、
- (f) 前記コラボレーション・サーバーから、個人情報を入力し、前記カスタマー側情報端末から前記コラボレーション・サーバーへ送信するためのカスタマー側制御ウインドウを生成し、カスタマーのログオン指示操作に応答して、カスタマーの個人情報および問い合わせ種別情報を含むログオン要求を前記カスタマー側情報端末から前記コラボレーション・サーバーに送信するカスタマー側第 1 アプレットと、
- (g) 前記問い合わせ種別情報に対応したエージェントを前記カスタマーにアサインするコール・マネージャーと、
- (h) 前記カスタマー側情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーからコンテンツ情報を取得するに際し、(1) 前記コンテンツ情報に前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出するカスタマー側第 2 アプレットと、(2) 変更後のコンテンツ情報を含むページの URL、前記変更後のコンテンツ情報を含むページの親のページの URL、前記親のページが持つ子のページの数、その親のページに対する前記変更後のコンテンツ情報を含むページの位置を含む変更後ページ情報を取得するカスタマー側スクリプトと、(3) 前記変化後ページ情報を前記コラボレーション・サーバに送信するとともに、エージェント側から送信された変化後ページ情報を取得したことに応答して、該送信された変化後ページ情報が現在カスタマー側ブラウザで表示さ

れているページに該当するか否か进行检查し、該当しない場合には、該送信された変化後ページ情報に対応したページを表示するように前記カスタマー側ブラウザに指示するツリー・マネージャーとを埋めこむキャッシュ・マネージャーと、

(i) コラボレーションを行なう前記エージェントと前記カスタマーのU A Iの組みを記録するとともに、前記カスタマー側第2アプレットが前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出したことに応答して、前記カスタマー側情報端末から送信された変更後ページ情報を前記エージェント側情報端末に送信するセッション・マネージャーと、

を含むコラボレーション・システム。

【請求項7】

コンテンツ情報を表示する第1のブラウザを搭載する第1の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第2の情報端末が取得することをサポートするコラボレーション・サーバーであって、

(a) コラボレーション・サーバーに接続する第1の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋め込むキャッシュ・マネージャーと、

(b) 前記アプレットが前記第1のブラウザが表示するコンテンツ情報の変更を検出したことに応答して前記第1の情報端末から送信された変更後のコンテンツ特定情報を前記第2の情報端末に送信するセッション・マネージャーと、

を含むコラボレーション・サーバー。

【発明の詳細な説明】

【0001】

【産業上の利用分野】

この発明は、情報処理方法に関し、詳しくは、遠隔地に存在する2以上の情報端末のブラウザを同期させる方法に関する。

【0002】

【従来の技術】

インターネット・コールセンターのエージェントとカスタマーのように、遠隔

地にある 2 つのブラウザを同期させる手法は、Corepoint Web Collaboration（「Corepoint Web Collaboration」は、IBM 社の商標）等の製品で実現されている。しかし、かかる技術においては、遠隔地にある 2 つの情報端末の夫々において、この機能を実現させるための特別のアプリケーションをインストールが必要であった。

【0003】

また、米Contigo社が取得した特許US5,944,791のようにサーバーでリンクを書き換え、かつアプレット等を使用してロードを制御するという手法もあるが、手入力したURLからリンク等のみしか同期できず、アプレットの内部的制御によるURLの変化等は検出できず、同期できないという問題があった。

【0004】

【発明が解決しようとする課題】

本発明の一つの目的は、予め情報端末にコラボレーション機能を提供する特別なソフトウェアを導入することなしに、遠隔地にある複数の情報端末がコラボレーションを行なうことを可能にするコラボレーション・システムを提供することにある。

【0005】

本発明の他の一つの目的は、情報端末で発生したイベントの如何を問わず、ウェブ・ブラウザの遷移を確実にコラボレーションを行なっている他の情報端末に伝達することのできるコラボレーション・システムを提供することにある。

【0006】

本発明の他の一つの目的は、コラボレーションを行うに際し、必要となる資源の削減された低コストのコラボレーション・システムを提供することにある。

【0007】

本発明の他の一つの目的は、コラボレーションを提供するソフトウェアの変更を即座に反映させることのできるコラボレーション・システムを提供することにある。

【0008】

本発明の他の一つの目的は、操作する情報端末のプラットフォームに依存する

ことのないコラボレーション・システムを提供することにある。

【0009】

【課題を解決するための手段】

本発明の一態様において、

カスタマー側情報端末に搭載されたブラウザとエージェント側情報端末に搭載されたブラウザは、コラボレーション・サーバー経由で接続されており、ウェブ・サーバーもこのコラボレーション・サーバー経由でアクセスするように制御されている。コラボレーション・サーバーは、カスタマー側のブラウザやエージェント側のブラウザがウェブ・サーバのあるページを参照するに際し、ページの変化を検出し、変更後のページ情報をコラボレーション・サーバー側に通知するアプレットと、コラボレーション・サーバ側から送信されたコラボレーションを行なう相手側の変更後のページ情報に対応したページを表示させるクライアント・コントローラーを埋めこむ。カスタマー側またはエージェント側ブラウザにおいて、ページの変化が検出されると、このアプレットが起動され、コラボレーション・サーバー経由で変更後のページ情報をコラボレーションを行なう相手側の情報端末に送られる。相手側のクライアント・コントローラーは、変更後のページを表示するようにブラウザを制御する。

【0010】

本発明の他の一態様においては、

コンテンツ情報を表示するカスタマー側ブラウザを搭載するカスタマー側情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報をエージェント側情報端末が取得するコンテンツ情報の取得方法であって、

(a) 前記エージェント側情報端末がコラボレーション・サーバーと通信接続する段階と、

(b) 前記エージェント側情報端末を一意に特定するU A I を生成する段階と、

(c) 前記コラボレーション・サーバーから、個人情報を入力し、前記エージェント側情報端末から前記コラボレーション・サーバーへ送信するためのエージェント側制御ウインドウを生成するエージェント側アプレットを取得する段階と

(d) 前記エージェント側アプレットを起動し、前記エージェント側制御ウィンドウを生成する段階と、

(e) エージェントのログオン指示操作に応答して、エージェントのユーザーIDを含むログオン要求を前記エージェント側情報端末から前記コラボレーション・サーバーに送信する段階と、

(f) 前記カスタマー側情報端末が前記コラボレーション・サーバーと通信接続する段階と、

(g) 前記カスタマー側情報端末を一意に特定するUAIを生成する段階と、

(h) 前記コラボレーション・サーバーから、個人情報を入力し、前記カスタマー側情報端末から前記コラボレーション・サーバーへ送信するためのカスタマー側制御ウィンドウを生成するカスタマー側第1アプレットを取得する段階と、

(i) 前記カスタマー側第1アプレットを起動し、前記カスタマー側制御ウィンドウを生成する段階と、

(j) カスタマーのログオン指示操作に応答して、カスタマーの個人情報および問い合わせ種別情報を含むログオン要求を前記カスタマー側情報端末から前記コラボレーション・サーバーに送信する段階と、

(k) 前記問い合わせ種別情報に対応したカスタマー待ち行列に該カスタマーのレコードを追加する段階と、

(l) 前記対応したカスタマー待ち行列における前記カスタマーのレコードの待ち状態を検査する段階と、

(m) 該検査結果をカスタマー側情報端末に通知する段階と、

(n) 前記問い合わせ種別情報に対応したエージェントを前記カスタマーにアサインする段階と、

(o) 前記エージェント側情報端末にセッションの開始を通知する段階と、

(p) 前記カスタマー側情報端末にセッションの開始を通知する段階と、

(q) 前記カスタマー側情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーからコンテンツ情報を取得するに際し、前記コラボレーション・サーバーが、前記コンテンツ情報に前記カスタマー側ブラウザが表示するコンテ

ンツ情報の変更を検出するカスタマー側第 2 アプレットを埋めこむ段階と、

(r) 変更後のコンテンツ情報を含むページの URL、前記変更後のコンテンツ情報を含むページの親のページの URL、前記親のページが持つ子のページの数、その親のページに対する前記変更後のコンテンツ情報を含むページの位置を含む変更後ページ情報を取得するカスタマー側スクリプトを埋めこむ段階と、

(s) 前記変化後ページ情報を前記コラボレーション・サーバに送信するとともに、エージェント側から送信された変化後ページ情報を取得したことに応答して、該送信された変化後ページ情報が現在カスタマー側ブラウザで表示されているページに該当するか否かを検査し、該当しない場合には、該送信された変化後ページ情報に対応したページを表示するように前記カスタマー側ブラウザに指示するツリー・マネージャーを埋めこむ段階と、

(t) 前記カスタマー側第 2 アプレットが前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出したことに応答して、変更後ページ情報を前記コラボレーション・サーバーに送信する段階と、

(u) 前記コラボレーション・サーバーが前記エージェント側情報端末に変更後ページ情報を送信する段階と、

(v) 前記エージェント側情報端末において、現在エージェント側ブラウザが表示中のページが変更後ページ情報に対応するか否か判断する段階と、

(w) 前記エージェント側ブラウザが現在表示中のページが変更後ページ情報に対応しないと判断された場合、前記エージェント側ブラウザに変更後ページ情報に対応するページをロードさせる段階と、

を含むコンテンツ情報の取得方法が提供される。

【 0 0 1 1 】

本願明細書の特許請求の範囲において、「ブラウザ」は、ネットスケープ・コミュニケーション（「ネットスケープ・コミュニケーション」は、ネットスケープ社の商標）やインターネット・エクスプローラー（「インターネット・エクスプローラー」は、マイクロソフト社の商標）等のウェブ・ブラウザや、アクロバット・リーダー（「アクロバット・リーダー」はアドビ社の商標）等のビューアーを含む概念である。また、「ウェブ・サーバー」は、インターネットを介してコン

テンツ情報をブラウザに提供するワールドワイドウェブサーバーや、ローカルのネットワークに接続し、ブラウザにコンテンツ情報を提供するサーバーを含む概念である。そして、「エージェント側アプレット」は、後述する実施例に示す「アプレット A」を含む概念である。「カスタマー側第 1 アプレット」は、後述する実施例に示す「アプレット C」を含む概念であり、「カスタマー側第 2 アプレット」は、後述する実施例に示す「アプレット P」を含む概念である。また、

(q) カスタマー側第 2 アプレットを埋めこむ段階、(r) カスタマー側スクリプトを埋めこむ段階、(s) ツリー・マネージャーを埋めこむ段階の各段階は、任意の順序で個別に行われても任意の組み合わせで同時に行われてもよい。

【 0 0 1 2 】

本発明の他の一態様においては、

コンテンツ情報を表示する第 1 のブラウザを搭載する第 1 の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第 2 の情報端末が取得するコンテンツ情報の取得方法であって、

(a) 前記第 1 の情報端末とコラボレーション・サーバーの間を通信接続する段階と、

(b) 前記第 2 の情報端末と前記コラボレーション・サーバーの間を通信接続する段階と、

(c) 前記第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋めこむ段階と、

(c) 前記第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信する段階と、

(e) 前記コラボレーション・サーバーが前記第 2 の情報端末に前記変更後のコンテンツ特定情報を送信する段階と、

(f) 前記第 2 の情報端末において、前記変更後のコンテンツ情報を取得する段階と、

を含むコンテンツ情報の取得方法が提供される。

【 0 0 1 3 】

なお、本願明細書の特許請求の範囲において、「クライアント・コントローラー」は、後述する実施例に説明する「ツリー・マネージャー」を含む概念であるが、フレームをサポートしないときのように、木構造の検出や比較の機能は必須ではなく、ブラウザに所定のコンテンツ情報を取得するよう指示する機能があればよい。

【 0 0 1 4 】

本発明の他の一態様においては、

コンテンツ情報を表示する第 1 のブラウザを搭載する第 1 の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第 2 の情報端末が取得するコンテンツ情報の取得方法であって、

(a) コラボレーション・サーバーに接続する第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋め込んで前記第 1 の情報端末に送信する段階と、

(b) 前記第 1 の情報端末の前記クライアント・コントローラーが送信した変更後のコンテンツ特定情報を前記コラボレーション・サーバーが受信する段階と、

(c) 前記コラボレーション・サーバーに通信接続された第 2 の情報端末において、前記変更後のコンテンツ情報を取得することを可能にするために、前記変更後のコンテンツ特定情報を前記第 2 の情報端末に送信する段階と、

を含むコンテンツ情報の取得方法が提供される。

【 0 0 1 5 】

本発明の他の一態様においては、

コンテンツ情報を表示する第 1 のブラウザを搭載する第 1 の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第 2 の情報端末が取得することをサポートするソフトウェア・プロダクトであって、

(a) コラボレーション・サーバーに接続する第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋め込むことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(b) 前記第 1 の情報端末から送信された変更後のコンテンツ特定情報を前記第 2 の情報端末に転送することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

を含むソフトウェアプロダクトが提供される。

【0016】

なお、本願明細書の特許請求の範囲において、「ソフトウェアプロダクト」とは、記憶媒体に格納されたソフトウェア（プログラムを包含する概念）や、ソフトウェア自体をも含む概念である。

【0017】

本発明の他の一態様においては、

コンテンツ情報を表示するカスタマー側ブラウザを搭載するカスタマー側情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報をエージェント側情報端末が取得することをサポートするソフトウェア・プロダクトであって

(a) コラボレーション・サーバーと通信接続するエージェント側情報端末を一意に特定する U A I を生成することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(b) 前記エージェント側情報端末において、個人情報を入力し、前記エージェント側情報端末から前記コラボレーション・サーバーへ送信するためのエージェント側制御ウィンドウを生成し、エージェントのログオン指示操作に応答して、エージェントのユーザー I D を含むログオン要求を前記エージェント側情報端末から前記コラボレーション・サーバーに送信するエージェント側アプレットと

(c) 前記コラボレーション・サーバーと通信接続するカスタマー側情報端末

を一意に特定するU A I を生成することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(d) 前記カスタマー側情報端末において、個人情報を入力し、前記カスタマー側情報端末から前記コラボレーション・サーバーへ送信するためのカスタマー側制御ウインドウを生成し、カスタマーのログオン指示操作に応答して、カスタマーの個人情報および問い合わせ種別情報を含むログオン要求を前記カスタマー側情報端末から前記コラボレーション・サーバーに送信するカスタマー側第1アプレットと、

(e) 前記問い合わせ種別情報に対応したエージェントを前記カスタマーにアサインすることを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(f) 前記エージェント側情報端末にセッションの開始を通知することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(g) 前記カスタマー側情報端末にセッションの開始を通知することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(h) 前記カスタマー側情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーからコンテンツ情報を取得するに際し、前記コラボレーション・サーバーが、前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出するカスタマー側第2アプレットを前記コンテンツ情報に埋めこむことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(i) 変更後のコンテンツ情報を含むページのURL、前記変更後のコンテンツ情報を含むページの親のページのURL、前記親のページが持つ子のページの数、その親のページに対する前記変更後のコンテンツ情報を含むページの位置を含む変更後ページ情報を取得するカスタマー側スクリプトを前記コンテンツ情報に埋めこむことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(j) 前記変化後ページ情報を前記コラボレーション・サーバに送信するとともに、エージェント側から送信された変化後ページ情報を取得したことに応答して、該送信された変化後ページ情報が現在カスタマー側ブラウザで表示されてい

るページに該当するか否かを検査し、該当しない場合には、該送信された変化後ページ情報に対応したページを表示するように前記カスタマー側ブラウザに指示するツリー・マネージャーを前記コンテンツ情報に埋めこむことを前記コラボレーション・サーバーに指示するためのプログラムコードと、

(k) 前記カスタマー側第2アプレットが前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出したことに応答して、前記カスタマー情報端末から送信された変更後ページ情報を前記エージェント側情報端末に送信することを前記コラボレーション・サーバーに指示するためのプログラムコードと、

を含むソフトウェアプロダクトが提供される。

【0018】

本発明の他の一態様においては、

コンテンツ情報を表示するカスタマー側ブラウザを搭載するカスタマー側情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報をエージェント側情報端末が取得することをサポートするコラボレーション・システムであって、

(a) カスタマー側ブラウザを搭載するカスタマー側情報端末と、

(b) エージェント側ブラウザを搭載するエージェント側情報端末と、

(c) 前記エージェント側情報端末前記カスタマー側情報端末との間の通信接続をサポートするHTTPブリッジと、

(d) 前記エージェント側ブラウザ及び前記カスタマー側ブラウザを夫々一意に特定するUAIを生成する段階と、

(e) 前記コラボレーション・サーバーから、個人情報を入力し、前記エージェント側情報端末から前記コラボレーション・サーバーへ送信するためのエージェント側制御ウインドウを生成し、エージェントのログオン指示操作に応答して、エージェントのユーザーIDを含むログオン要求を前記エージェント側情報端末から前記コラボレーション・サーバーに送信するエージェント側アプレットと

(f) 前記コラボレーション・サーバーから、個人情報を入力し、前記カスタマー側情報端末から前記コラボレーション・サーバーへ送信するためのカスタマ

一側制御ウィンドウを生成し、カスタマーのログオン指示操作に応答して、カスタマーの個人情報および問い合わせ種別情報を含むログオン要求を前記カスタマー側情報端末から前記コラボレーション・サーバーに送信するカスタマー側第1アプレットと、

(g) 前記問い合わせ種別情報に対応したエージェントを前記カスタマーにアサインするコール・マネージャーと、

(h) 前記カスタマー側情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーからコンテンツ情報を取得するに際し、(1) 前記コンテンツ情報に前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出するカスタマー側第2アプレットと、(2) 変更後のコンテンツ情報を含むページのURL、前記変更後のコンテンツ情報を含むページの親のページのURL、前記親のページが持つ子のページの数、その親のページに対する前記変更後のコンテンツ情報を含むページの位置を含む変更後ページ情報を取得するカスタマー側スクリプトと、(3) 前記変化後ページ情報を前記コラボレーション・サーバに送信するとともに、エージェント側から送信された変化後ページ情報を取得したことに応答して、該送信された変化後ページ情報が現在カスタマー側ブラウザで表示されているページに該当するか否かを検査し、該当しない場合には、該送信された変化後ページ情報に対応したページを表示するように前記カスタマー側ブラウザに指示するツリー・マネージャーとを埋めこむキャッシュ・マネージャーと、

(i) コラボレーションを行なう前記エージェントと前記カスタマーのU A Iの組みを記録するとともに、前記カスタマー側第2アプレットが前記カスタマー側ブラウザが表示するコンテンツ情報の変更を検出したことに応答して、前記カスタマー側情報端末から送信された変更後ページ情報を前記エージェント側情報端末に送信するセッション・マネージャーと、

を含むコラボレーション・システムが提供される。

【0019】

本発明の他の一態様においては、

コンテンツ情報を表示する第1のブラウザを搭載する第1の情報端末がウェブ・サーバーから取得したコンテンツ情報と同様な情報を第2の情報端末が取得す

ることをサポートするコラボレーション・サーバーであって、

(a) コラボレーション・サーバーに接続する第 1 の情報端末が前記コラボレーション・サーバー経由でウェブ・サーバーから新たなコンテンツ情報を取得するに際し、変更後のコンテンツ特定情報を前記コラボレーション・サーバーに送信するクライアント・コントローラーを埋め込むキャッシュ・マネージャーと、

(b) 前記アプレットが前記第 1 のブラウザが表示するコンテンツ情報の変更を検出したことに応答して前記第 1 の情報端末から送信された変更後のコンテンツ特定情報を前記第 2 の情報端末に送信するセッション・マネージャーと、

を含むコラボレーション・サーバーが提供される。

【実施例】

【0020】

A. 概要

図 1 は、本発明の好適な実施例におけるコラボレーションシステムの概要を示す図である。図 1 に示すように、カスタマーは、コラボレーション・サーバー 110 を経由してウェブ・サーバー 140 アクセスする。コラボレーション・サーバー 110 は、そのカスタマーのブラウザを一意に特定する ID (UAI) をアサインし、その ID の組み合わせでコラボレーションセッションを確立する。このときにそのカスタマーに応答するエージェントの割り当てが行われ、エージェントとカスタマー間のセッションが確立する。

【0021】

コラボレーション・サーバー 110 では、BODY タグを含む html に対し、html の構造及びその変化を検出するスクリプト、アプレットを埋めこむ変更を行なう。この変更によって、その html のロードが完了した時点で、JavaScript の function が起動される。この JavaScript の function は、現在のウィンドウから最上位のウィンドウまでの URL と、各ノードにおけるウィンドウの番号情報を取得する。

【0022】

エージェント側及びクライアント側で起動された各 JavaScript は、自分とその親の情報 (自分の URL、親の URL、親が持つ子の数、その親に対する自分の

位置)を取得し、各htmlに挿入されたアプレットに情報を渡す。アプレットは、現在表示しているhtmlの構造を管理するツリー管理クラスにその情報を引き渡す。なお、このツリー管理クラスはJava VM内でGUIを持たないスレッドとして実行され、各htmlのアプレットから共通に参照される。

【0023】

ツリー管理クラスでは、ツリーに変更が発生する度にコラボレーションの相手にツリー構造を送信する。この通信を行うクラスもGUIを持たないスレッドとして実行している。ツリー情報を受信した場合には、現在の自分のツリーと比較し、相違点があり、自分が遷移開始側でなければ、相違を修正するようにロードを開始する。

【0024】

B. ハードウェア構成

図2を参照すると、本発明において使用されるコラボレーション・サーバー110のハードウェア構成の一実施例を示す概観図が示されている。コラボレーション・サーバー110は、中央処理装置(CPU)1とメモリ4とを含んでいる。CPU1とメモリ4は、バス2を介して、補助記憶装置としてのハードディスク装置13とを接続してある。フロッピーディスク装置(またはMO、CD-ROM等の記録媒体駆動装置26、28、29)20はフロッピーディスクコントローラ19(またはIDEコントローラ25、SCSIコントローラ27等)を介してバス2へ接続されている。

【0025】

フロッピーディスク装置(またはMO、CD-ROM等の記録媒体駆動装置)20には、フロッピーディスク(またはMO、CD-ROM等の記録媒体)が挿入され、このフロッピーディスク等やハードディスク装置13、ROM14には、オペレーティングシステムと協働してCPU等に命令を与え、本発明を実施するためのコンピュータ・プログラムのコードを記録することができ、メモリ4にロードされることによって実行される。このコンピュータ・プログラムのコードは圧縮し、または、複数に分割して、複数の媒体に記録することもできる。

【 0 0 2 6 】

コラボレーション・サーバー 1 1 0 は更に、ユーザ・インターフェース・ハードウェアを備え、入力をするためのポインティング・デバイス（マウス、ジョイスティック等）7 またはキーボード 6 や、視覚データをユーザに提示するためのディスプレイ 1 2 を有することができる。タッチ・パネルを入力手段とすることもできる。また、パラレルポート 1 6 を介してプリンタを接続することや、シリアルポート 1 5 を介してモデムを接続することが可能である。このコラボレーション・サーバー 1 1 0 は、シリアルポート 1 5 およびモデムまたは通信アダプタ 1 8（イーサネットやトークンリング・カード）等を介してネットワークに接続し、他のコンピュータ等と通信を行うことが可能である。

【 0 0 2 7 】

スピーカ 2 3 は、オーディオ・コントローラ 2 1 によって D/A（デジタル／アナログ変換）変換された音声信号を、アンプ 2 2 を介して受領し、音声として出力する。また、オーディオ・コントローラ 2 1 は、マイクロフォン 2 4 から受領した音声情報を A/D（アナログ／デジタル）変換し、システム外部の音声情報をシステムにとり込むことを可能にしている。

【 0 0 2 8 】

このように、本発明のコラボレーション・サーバー 1 1 0 は、通常のパーソナルコンピュータ（PC）やワークステーション、ノートブック PC、パームトップ PC、ネットワークコンピュータ等を含む通信機能有する情報端末、または、これらの組合せによって実施可能であることを容易に理解できるであろう。ただし、これらの構成要素は例示であり、その全ての構成要素が本発明の必須の構成要素となるわけではない。

【 0 0 2 9 】

特に、ここで説明したハードウェア構成は、情報端末のコラボレーション支援に必須でないものもあるため、音声の処理に必要なオーディオ・コントローラ 2 1、アンプ 2 2、スピーカ 2 3、マイクロフォン 2 4 やオペレータからの直接的な入力を可能にするキーボード 6、マウス 7、キーボード／マウスコントローラ 5、視覚データをユーザに提示するための CRT 1 2、表示装置 1 1、VRAM

9、VGA 8、各種記録媒体処理装置 1 9、2 5、2 7 等はなくともよい。

【0030】

このコラボレーション・サーバー 1 1 0 の各構成要素は、複数のマシンを組合せ、それらに機能を配分し実施する等の種々の変更は当業者によって容易に想定されるものであって、それらの変更は、本発明の思想に包含される概念である。

【0031】

本発明に使用されるクライアント側情報端末 1 3 0、エージェント側情報端末 1 7 0 もコラボレーション・サーバー 1 1 0 と同様に図 2 に示すハードウェア構成によって実施することができる。すなわち情報端末 1 3 0 も、情報入手要求を入力し、それを発信・受信する機能とがあればよいので、通常のパーソナルコンピュータ（PC）やワークステーション、ノートブック PC、パームトップ PC、コンピュータを内蔵したテレビ等の各種家電製品、通信機能を有するゲーム機、電話、FAX、携帯電話、PHS、電子手帳、等を含む通信機能有する情報端末、または、これらの組合せによっても実施可能であることを容易に理解できるであろう。ただし、これらの構成要素は例示であり、その全ての構成要素が本発明の必須の構成要素となるわけではない。

【0032】

コラボレーション・サーバー 1 1 0 側のオペレーティング・システムとしては、Windows NT（マイクロソフトの商標）、Windows 9 x（マイクロソフトの商標）、Windows 3. x（マイクロソフトの商標）、OS/2（IBMの商標）、Mac OS（Apple 社の商標）、Linux（Linus Torvalds の商標）、AIX（IBMの商標）上の X-WINDOW システム（MIT の商標）などの、標準で GUI マルチウインドウ環境をサポートするものや、PC-DOS（IBMの商標）、MS-DOS（マイクロソフトの商標）などのキャラクタ・ベース環境のもの、さらには OS/Open（IBM の商標）、VxWorks（Wind River Systems, Inc. の商標）等のリアルタイム OS、Java OS 等のネットワークコンピュータに組み込まれた OS でも実現可能であり、特定のオペレーティング・システム環境に限定されるものではない。

【0033】

クライアント側情報端末130、エージェント側情報端末170のオペレーティング・システムも、Windows NT（マイクロソフトの商標）、Windows 9x（マイクロソフトの商標）、Windows 3.x（マイクロソフトの商標）、OS/2（IBMの商標）、Mac OS（Apple社の商標）、Linux（Linus Torvaldsの商標）、AIX（IBMの商標）上のX-WINDOWシステム（MITの商標）などの、標準でGUIマルチウインドウ環境をサポートするものや、PC-DOS（IBMの商標）、MS-DOS（マイクロソフトの商標）などのキャラクタ・ベース環境のもの、さらにはOS/Open（IBMの商標）、VxWorks（Wind River Systems, Inc.の商標）等のリアルタイムOS、JavaOS等のネットワークコンピュータに組み込まれたOSでも実現可能であり、特定のオペレーティング・システム環境に限定されるものではない。

【0034】

C. システム構成

図3は、エージェントがログオンするに際し動作する処理要素を含む本発明の好適な実施例におけるコラボレーション・システムのシステム構成を示す機能ブロック図である。図4は、エージェントとカスタマーがコラボレーションを開始した際に動作する処理要素を含む本発明の好適な実施例におけるコラボレーション・システムのシステム構成を示す機能ブロック図である。図5は、クライアント側情報端末の処理要素を示す機能ブロック図である。図6は、カスタマー側情報端末でURLの変更を伴うイベントが発生した際に動作する処理要素を含む本発明の好適な実施例におけるコラボレーション・システムのシステム構成を示す機能ブロック図である。

【0035】

本発明の好適な実施例におけるコラボレーション・システム100は、カスタマー側情報端末130、コラボレーション・サーバー110、ウェブ・サーバー140、コールマネジャー160、キューサーバー180、エージェント側情報

端末170を含んでいる。また、本発明の好適な実施例におけるコラボレーション・サーバー110は、キャッシュ・マネージャー115、HTTPブリッジ111、セッション管理部113、セッション・マネージャー117、UAI生成部119、アプレットA151、アプレットC153、ツリー・マネージャー155、アプレットP157を含んでいる。

【0036】

本発明の好適な実施例におけるコラボレーション・サーバー110は、Webサーバーとしての機能も持ち、エージェント用のアプレットA、カスタマー用のアプレットC、ページ代表アプレットPとPが参照するツリー・マネージャー、通信関係のクラスといったJavaのクラス類とスクリプトを保管している。これらは、キャッシュ・マネージャーによって処理されない特別なディレクトリーに置かれており、これらに対する要求は、コラボレーション・サーバー自体がWebサーバーとして処理する。なお、クライアント側情報端末130のブラウザ131とエージェント側情報端末170のブラウザ171の間でセッションを確立する技術は、Corepoint Web Collaboration（「Corepoint」は、IBM社の商標）の機能を利用することによって実現することができる。

【0037】

HTTPブリッジ111は、セッション・マネージャー117とソケット接続し、ブラウザ131、171との間の通信をサポートする。HTTPブリッジでは、図11に示すUAI-ソケット管理テーブル280を保持しており、ブラウザ側からの接続毎にエントリを追加する。

【0038】

セッション管理部113は、エージェントからのログオン要求を取得し、コール・マネージャー160に伝達する。また、コール・マネージャー160からエージェントのログオン処理が正常に終了した旨の通知を受け、これをエージェント側に通知する。

【0039】

キャッシュ・マネージャー115は、ウェブ・ブラウザ140が送信したHTTPのリクエストに対するHTMLのコンテンツ（HTTPのレスポンス）をキ

キャッシュする。また、アプレットやスクリプトの埋め込む処理を行なう。

【0040】

セッション・マネージャー117は、コラボレーションを行なうエージェントとカスタマーの組をコール・マネージャー160から受け取り、図7のセッション管理テーブルの空きエントリを検索して空きのセッション番号を割り当て、エージェントとカスタマーのUAIを記録する。割り当てたセッション番号は、コール・マネージャーに返される。また、セッション・マネージャー117はUAIの組をキャッシュ・マネージャー115に通知する。これを受けたキャッシュ・マネージャー115は、図10に示すキャッシュ管理テーブル270でどの要求とどの要求を対応させるかを管理する。

【0041】

UAI生成部119は、コラボレーションの参加者を一意に特定するためのIDを生成し、それをブラウザのCookieとして設定する。

【0042】

アプレットC135、153は、制御ウィンドウ133を生成する。また制御ウィンドウ133においてユーザーが入力した情報とUAIを含む、エージェント割り当て要求を、コラボレーション・サーバー上にあるセッション管理部113に伝達する。また、アプレットC135、153は、すべてのエージェントがサービス中であった場合には、待ち行列内の位置を取得して、その情報を制御ウィンドウ133に渡す。

【0043】

アプレットA151、175は、制御ウィンドウ173を生成し、制御ウィンドウ173にエージェントが入力した情報とUAIを含むログオン情報をコラボレーション・サーバー上にあるセッション管理サブレット113に伝達する。

【0044】

エージェント側制御ウィンドウ173は、エージェントID、待ち行列名（スキル・グループ）などを入力して実際にログオンするためのインターフェースを提供する。待ち行列名は、紳士服担当、婦人服担当といったようにエージェント毎に担当業務が異なる場合に、エージェントが処理することができる要求の種類

を示すもので、コール・センターでは一般にスキル・グループとして知られる概念である。なお、スキル・グループの実装は必須ではなく、省略可能であり、その場合には待ち行列名を固定とすれば良い。また、エージェント側制御ウィンドウ 1 7 3 は、現在のエージェントの状態（カスタマーの接続待機状態等）を示す表示を行なう。

【 0 0 4 5 】

本発明の好適な実施例において、情報端末 1 3 0、1 7 0 にはウェブ・ブラウザ 1 3 1、1 7 1 が搭載されている。ウェブ・ブラウザ 1 3 1、1 7 1 は、URL を指定し、所定のウェブ・サーバーへ向けリクエストを送信する。また、ウェブ・サーバー 1 4 0 から送信されたレスポンスを受領し、表示画面上にそれを表示する。サーバー 1 1 0、1 4 0 から取得したアプレット等を実行することもできる。この図において、説明の便宜上、ウェブ・サーバー 1 4 0 は単一のサーバーとして描かれているが、ウェブ・ブラウザ 1 3 1、1 7 1 は、種々のウェブ・サーバーに対し h t m l ファイル取得要求を送信することができる。

【 0 0 4 6 】

なお、本発明の好適な実施例においては、h t m l がフレームを含むものである場合には、各フレームを構成する h t m l 毎に部分木構造を検出するスクリプトと、ページを代表するアプレットが埋め込まれ、ツリー構造を検出するためのクラスと通信のためのクラスがブラウザによってロードされる。これらのクラスは、ブラウザの Java Virtual Machine 内で動作し、ツリー構造を検出するためのクラスと通信のためのクラスはフレームを構成する h t m l の数によらず、常に 1 組だけ存在する。

【 0 0 4 7 】

このため、本発明では、ブラウザは以下の 3 つの条件を満たすものであることが必要である。

- Java が使用可能
- Script (通常は JavaScript) が使用可能
- 上記 Script から Java のメソッドの呼び出しが可能

この 3 つの条件を満たすものとしては、Netscape Communicator 4.5x 以降、Micr

Microsoft Internet Explorer 4.01以降等があげられる。なお、W3C (World Wide Web Consortium) で規格策定中である、DOM(Document Object Model) において十分な機能が規定され、Javaバインディングも規定され、これがブラウザで実装された時点では、スクリプトを使用しない実装も可能となる。また、現時点においても、フレームを使用したhtmlをサポートしないと限定すれば、スクリプトをサポートしないブラウザにおいても実施可能である。

【0048】

カスタマー側の制御ウィンドウ133は、カスタマーが名前や電話番号などの個人情報や質問の種別や内容を入力するためのインターフェースを提供する。また、待ち行列の状態等、そのカスタマーの状況を示す情報を表示する。

【0049】

ツリー・マネージャー137、155、177は、ブラウザ131、171に表示されるページのツリー構造を解析する。通信部139は、HTTPブリッジ111を介してセッション・マネージャー117と通信する。スクリプト157は、自分とその親の情報（自分が埋めこまれたページのURL、親のURL、親が持つ子の数、その親に対する自分の位置）を取得し、各htmlに挿入されたアプレットに情報を渡す。

【0050】

コール・マネージャー160は、図8に示すようなエージェント管理テーブル230を保持しており、エージェントがログオンすると、その状態を更新し、エージェントにアサインされているUAIも設定する。また、コール・マネージャー160は、図9に示すカスタマー待ち行列250も管理しており、カスタマーからのコラボレーション要求があると、エントリーを作成して、キュー・サーバー180の待ち行列に入れる。キュー・サーバー180は、待ち行列名に対応したエージェントが待機状態であれば、これを割り当て、待機状態のエージェントが無い場合には、待ち行列に入れる。

【0051】

このキューサーバー180やコール・マネージャー160は、コール・センターと統合するかどうかといったシステムの実装要件に応じて、本願の出願時にお

いて利用可能なさまざまな製品によって実現することができる。例えば、コラボレーション・サーバー 110 として Corepoint Web Collaboration を採用した場合、Corepoint Telephony (「Corepoint Telephony」は、IBM 社の商標) と統合するにはバーチャル ACD を採用し、また単体で使用するためには、ソフトウェア ACD および イージー ACD を採用することにより実現することができる。Corepoint Telephony と統合した場合には、ブラウザによるコラボレーション (同期処理) と同時に、電話回線を通じてカスタマーはエージェントと会話することができる。

【0052】

エージェント割当てを管理するキューサーバー 180 やコール・マネージャー 160 等のコンポーネントは、コラボレーション・サーバー 110 上に置くことも可能であるが、図 3、4、6 のように、別サーバー上に置いて、コラボレーション・サーバー 110 内には、このサーバーとの通信を行うクライアントだけを置くことにより、エージェント割り当てを行うサーバーをファイア・ウォールの内側に置き、セキュリティ性の高いシステムを構築することができる。

【0053】

以上図 3～6 に示す各機能ブロックを説明したが、これらの機能ブロックは、論理的機能ブロックであり、各々 1 つのまとまりを持ったハードウェアやソフトウェアによって実現されることを意味するのではなく、複合し、または共通したハードウェアやソフトウェアによって実現可能である。また、この図 3 に示す全ての機能ブロックが本発明に必須の構成要素という訳ではない。

【0054】

例えば、エージェント割当ての機能をコラボレーション・サーバー 110 に持たせることにより、コール・マネージャー 160 やエージェント割当てサーバー 160 は不要なものとなる。

【0055】

D. 動作手順

D-1. エージェントのログオン

図16は、エージェントのログオンの手順を示すフローチャートである。まず、エージェントがブラウザ171に所定URLを入力する等により、コラボレーション・サーバー110にアクセスすることにより図に示す手順は開始する（ステップ403）。ログオンを指示するページを表示させると、そのページにはページ代表アプレットPを埋め込むAPPLETタグがあり、それによってアプレットPとツリー・マネージャー、通信関連のクラス177がコラボレーション・サーバー110から取得される（ステップ403）。これにより、後述の同期処理において、スクリプトがアプレットPを呼び出そうとした時点でアプレットP、ツリー・マネージャー、通信関連のクラス177がブラウザのキャッシュに置かれ、直ちに使用可能となる。具体的には、以下のようにHTMLを変換することにより、埋め込みを行なう。

【0056】

変更1（BODYタグ部分）

```
<BODY onload="CWCLoadF()" onResize=".....">
<applet code="CWC001.class" name="CWCApp1" codebase="/Tree/tu" align="right" hspace="0" vspace="0" width="0" height="0">
</APPLET>
```

【0057】

変更2（/BODYタグの直前）

```
<script language="JavaScript">
Window.setTimeout("CWCLoadF()", 1000);
function CWCLoadF() {.....
</script>
</body>
```

【0058】

一部のブラウザの、ブラウザウィンドウの大きさを変更しただけでhtmlの再ロードが生じ、かつJavaScriptのfunctionの呼び出しに失敗するという障害を回避するために、bodyタグのonResizeには、functionの内容を直接記述する。また、

別の障害を回避するため、/bodyタグの直前に、スクリプトをhtml内に埋め込む。なお、実際のアプレットタグのコードベースパラメータはサーバーの構成によって異なる。また、クラス名には実施企業名を含むパッケージ名を含む。この変更によって、onloadの指定により、ロードが完了した時点で、JavaScriptのfunctionが起動される。このJavaScriptのFunctionは、現在のウィンドウから最上位のウィンドウまでのURLと、各ノードにおけるウィンドウの番号情報を取得する。取得する情報の例を図12、13で示す。この例では、abc/def.htmにうめこまれたスクリプトが取得する情報を示している。なお、一部のブラウザではロード中にストップボタンを押すなどしてロードを中止した場合にはonloadイベントが発生しないため、スクリプトではタイマーによってJavaScriptのfunctionが起動されるように設定する。場合によっては2回実行されてしまうが、アプレットで先に来た方の情報を使用することで、中止時にonloadが発生しないブラウザに対応でき、かつ、ページ内のイメージのロードに時間がかかる場合にはそのロード終了を待たずにツリー構造が取得できるようにしている。

【0059】

エージェントがログオンボタンをクリックしてログオン処理の開始を要求すると、その要求はUAI生成サブレット119に送られる（ステップ405）。UAI生成サブレット119は、本発明のエージェント用のUAI（User Access Identifier）を生成し、UAIとコラボレーションログオンページへのリダイレクションを返送する（ステップ407）。UAIはコラボレーションの参加者を一意に特定するためのIDであり、ブラウザのCookieとして設定される。また、ここではリダイレクションを用いているが、特定のページを表示させる手法は、METAタグによる遷移等の手法があり、リダイレクションの使用は本発明の実施に必須の要件ではない。

【0060】

ブラウザ171は、リダイレクションの指示に基づいて、キャッシュ・マネージャー115経由でログオンページを取得する（ステップ409）。この際、UAIが設定されているため、キャッシュ・マネージャー115は、予め定められたウェブ・サーバー140からページを取得し、アプレットPとスクリプトを埋

め込むように変更を行った後、エージェント側のブラウザ 1 7 1 に返送する（ステップ 4 1 1）。ログオンページには、アプレット A 1 5 1 を指定したAPPLETタグが含まれている。

【 0 0 6 1 】

エージェント側のブラウザ 1 7 1 は、アプレットタグの指示により、アプレット A 1 5 1 を取得する（ステップ 4 1 3）。なお、本発明の好適な実施例においては、ここで取得されるHTMLにもアプレット P 1 7 7 を指定したAPPLETタグもキャッシュ・マネージャー 1 1 5 によって埋め込まれているが、すでにステップ 4 0 3 において取得済みであるため、この例においては実際の取得は行われない。

【 0 0 6 2 】

ブラウザ 1 7 1 によってアプレット A 1 7 5 が起動されると、アプレット A 1 7 5 は、制御ウィンドウ 1 7 3 を表示する（ステップ 4 1 7）。制御ウィンドウ 1 7 3 は、実際にログオンのリクエストを送信する際に、エージェント ID、待ち行列名（スキル・グループ）などを入力するために提供されたユーザーインターフェースである。待ち行列名は、紳士服担当、婦人服担当といったようにエージェント毎に担当業務が異なる場合に、エージェントが処理することができる要求の種類を示すもので、コール・センターでは一般にスキル・グループとして知られる概念である。なお、スキル・グループの実装は必須ではなく、省略可能であり、その場合には待ち行列名を固定とすれば良い。

【 0 0 6 3 】

エージェントが制御ウィンドウ 1 7 3 においてログオンを指示するボタンをクリックすると（ステップ 4 1 9）と、アプレット A 1 7 5 は、エージェントが入力した情報と U A I を含むログオン情報をコラボレーション・サーバー 1 1 0 上にあるセッション管理サブレット 1 1 3 に伝達する（ステップ 4 2 1）。セッション管理サブレット 1 1 3 は、その情報をコール・マネージャー 1 6 0 に伝達する（ステップ 4 2 3）。コール・マネージャー 1 6 0 は、その情報をキュー・サーバー 1 8 0 に伝達する。これを受領したキュー・サーバー 1 8 0 は、図 8 のエージェント管理テーブル 2 3 0 のエージェントの状態を更新し、エージェン

トにアサインされているU A I も設定する。具体的には、エージェント I D 2 3 1 と U A I 2 3 7 を登録し、待ち行列名 2 3 3 には、そのエージェントのスキルグループを、状態 2 3 5 には待機状態である旨の情報をセットする。

【 0 0 6 4 】

なお、この実施例では、U A I の生成機能とセッション管理は、サーブレットとして実現されているが、クライアントを一意に特定する機能があればよいので、CGIその他のすでに知られているWebサーバーアプリケーション実装方法によっても実現可能である。また、本発明の好適な実施例において、キュー・サーバー 1 8 0 は、カスタマーの待ち行列を管理する機能を有しているが、コール・センターと統合するかどうかといったシステムの実装要件に応じて、さまざまなキュー・サーバー 1 8 0 を選択することができる。

【 0 0 6 5 】

そして、セッション管理サーブレット 1 1 3 は、コール・マネージャーからエージェントのログオン処理が正常に終了した旨の通知を受け、これをアプレット A 1 7 5 に通知する。アプレット A 1 7 5 は、カスタマーの接続待機状態であることを示す表示を制御ウインドウ 1 7 3 に表示し、待機する（ステップ 4 2 5、4 2 7）。コラボレーション・サーバー 1 1 0 のセッション管理サーブレット 1 1 3 も待機する（ステップ 4 2 9）

【 0 0 6 6 】

D-2. カスタマーのログオンとコラボレーションの開始

図 1 7、1 8 は、カスタマーがログオンし、エージェントとコラボレーションの開始する手順を示すフローチャートである。このフローの初期状態において、ユーザーのブラウザは、ウェブページが格納されている任意のウェブ・サーバーにアクセスしている（ステップ 4 5 1）。そして、カスタマーはURLを入力する等、所定のURLを指定することにより、エージェントとのコラボレーションを開始するボタンを有するページに誘導するページを有する所定のウェブサーバー 1 4 0 にアクセスする（ステップ 4 5 3）。本発明の好適な実施例において、このようにコラボレーションサーバー 1 1 0 とウェブサーバー 1 4 0 を分けた形

態をとることによって、コラボレーションを必要としないユーザーが増えても、ウェブサーバー 1 4 0 のみを使用されることになるので、コラボレーション・サーバー 1 1 0 の負荷が増加しない。ただし、Corepoint Web Collaborationの既存のクライアント・コードを利用した場合と互換性を持つように、カスタマーに予めコラボレーション・サーバー経由でアクセスを行わせるという実施形態も可能である。

【 0 0 6 7 】

カスタマーが呼び出しボタンのあるページを表示すると、そのページには U A I が設定されていない場合には呼び出しボタンを作成して、ページ代表アプレット P を埋め込む APPLLET タグが生成するスクリプトが含まれており、このスクリプトによって APPLLET タグが生成され、アプレット P と ツリー・マネージャー、通信関連のクラスがコラボレーション・サーバー 1 1 0 から取得される（ステップ 4 5 5）。

【 0 0 6 8 】

なお、本発明の好適な実施例においては、カスタマーが呼び出しボタンをクリックしてコラボレーションの開始を要求する際、要求には、呼び出しもとのページを識別するホスト名部分を除いた URL が含まれている。この要求は U A I 生成サーブレット 1 1 9 に送られる（ステップ 4 5 7）。U A I 生成サーブレット 1 1 9 は、本発明のカスタマー用の U A I（User Access Identifier）を生成し、U A I とコラボレーションログオンページ（実際には、呼び出しボタンがあるページをキャッシュ・マネージャー経由で取得したもの）へのリダイレクションを返送する（ステップ 4 5 9）。U A I はコラボレーションの参加者を一意に特定するための I D である。

【 0 0 6 9 】

ブラウザ 1 3 1 は、リダイレクションの指示に基づいて、キャッシュ・マネージャー 1 1 5 経由で初期表示ページを取得する。（ステップ 4 6 1）この際、U A I が設定されているため、キャッシュ・マネージャー 1 1 5 は予め定められた Web サーバー 1 4 0 からページを取得し、アプレット P 1 5 7 とスクリプトを埋め込むように変更を行った後、カスタマー側のブラウザ 1 3 1 に返送する（ステ

ップ463)。初期表示ページには、コラボレーション・アプリケーションである、アプレットCを指定したAPPLETタグが含まれている。

【0070】

カスタマー側のブラウザは、アプレットタグの指示により、アプレットCを取得する（ステップ465）。ここでもアプレットPを指定したAPPLETタグもキャッシュ・マネージャーによって埋め込まれているが、すでにステップ455において取得済みであるため、この例において実際の取得は行われない（ステップ467）。

【0071】

ブラウザによってアプレットC135が起動されると、アプレットC135は制御ウィンドウ133を表示する（ステップ469）。なお、制御ウィンドウ133をブラウザウィンドウで実現する形態、制御ウィンドウとして独立したウィンドウを表示せず、カスタマーのブラウザ内にhtmlとアプレットで制御ウィンドウの機能を実現し、制御ウィンドウへの情報の入力後に、本来のWebページへ切り替えるという形態も可能である。

【0072】

カスタマー側に表示される制御ウィンドウ133は、カスタマーが名前や電話番号などを入力して実際にログオンするために提供されるインターフェースである。エージェントのスキルグループが設定されている場合には、この時点でカスタマーが選択できるようにするか、あるいは呼び出しボタンのあるページに対応して自動的に設定するようにする。

【0073】

カスタマーが所定の情報を入力してエージェントとのコラボレーションの開始を要求する（ステップ471）と、アプレットC135は、ユーザーが入力した情報とUAIを含む、エージェント割り当て要求を、コラボレーション・サーバー110上にあるセッション管理サブレット113を経由して、コール・マネージャー160に伝達する（ステップ473）。

【0074】

コール・マネージャー160は、図9に示すカスタマー待ち行列のエントリー



250を作成して、キュー・サーバー180の待ち行列に入れる。(ステップ475)。キュー・サーバー180は、待ち行列名に対応したエージェントが待機状態であれば、これを割り当て、待機状態のエージェントが無い場合には、待ち行列に入れる。アプレットC135は、すべてのエージェントがサービス中であった場合には、待ち行列内の位置を取得して、その情報を表示する(ステップ481)。

【0075】

エージェントが割り当てられると、コール・マネージャー160は、エージェントとカスタマーの組をセッション・マネージャー117に通知する(ステップ485、487)する。なお、図17の例では、カスタマー側のアプレットC135が定期的にセッション管理部113経由でコール・マネージャー160に問い合わせ、待ち行列の変化を監視できるようになっている(ステップ477~483)。

【0076】

エージェントが割り当てられ、コール・マネージャー160からエージェントとカスタマーの組の情報を受けると、セッション・マネージャー117は、図7のセッション管理テーブル210の空きエントリを検索して空きのセッション番号を割り当て、エージェントとカスタマーのUAIを記録する。さらに、セッション・マネージャー117はUAIの組をキャッシュ・マネージャー115に通知し、キャッシュ・マネージャー115は、図10に示すキャッシュ管理テーブル270でどの要求とどの要求を対応させるかを管理する。

【0077】

セッション・マネージャー117は、セッション番号をコール・マネージャー160に返す。コール・マネージャー160は、エージェント割り当ての完了とセッション番号を、セッション管理サブレット113に通知する(ステップ487)。この通知は、セッション管理サブレット113からエージェント側(ステップ489)、カスタマー側(ステップ495)それぞれに通知する。

【0078】

セッション管理サブレット113は、エージェントが割り振られたこととセ

セッション番号をアプレットA175に通知し（ステップ487、489）、アプレットA175は、カスタマー側のブラウザ131内のツリー・マネージャーとの通信を確立するために、セッション番号とUAIを指定して通信クラス139に対して通信の開始を要求する（ステップ491）。なお、テキスト・チャットを行う場合には、エージェントの名前またはニックネームも通信の開始時に指定する。通信クラス139は、HTTPブリッジ111を介してセッション・マネージャー117との通信を開始する（ステップ493）。

【0079】

HTTPブリッジ111とセッション・マネージャー117の間はブラウザ毎にソケット接続を行う。HTTPブリッジ111では、図11に示すUAI-ソケット管理テーブル280を保持しており、接続毎にエントリを追加する。セッション・マネージャー117は、httpブリッジ111とのセッションが確立すると、セッション管理テーブル210のソケット情報を更新する。

【0080】

一方、セッション管理サブレット113は、エージェントが割り振られたこととセッション番号をカスタマー側のアプレットC135に通知（ステップ487、495）し、アプレットC135は、エージェント側のブラウザ内のツリー・マネージャー177との通信を確立するために、セッション番号を指定して通信クラス139に対して通信の開始を要求する（ステップ497）。通信クラス139は、HTTPブリッジ111を介してセッション・マネージャー117との通信を開始する（ステップ497、499）。セッション・マネージャー117は、httpブリッジ111とのセッションが確立すると、セッション管理テーブル210のソケット情報を更新する。

【0081】

なお、本発明の好適な実施例において、通信クラス139とHTTPブリッジ111間の通信は、JavaのURLConnectionを利用して行うことにより、コラボレーション・サーバーとブラウザの間にプロキシあるいはソックスといったファイアーウォールが設置されている場合でも通信を可能としているが、ファイアーウォールを超えることが要件でない場合には、Socketを使用して実装することも可能

である。

【0082】

セッション・マネージャー117は、セッションを構成する2つのソケット接続を受け付けると、通信が開始されたことを通知する（ステップ501）。これを受けて、アプレットA175とアプレットC135は、通信が開始されたことを表示（ステップ503、505）し、アプレットC135は同期の開始を要求する（ステップ507）。これは、カスタマー側に表示されているページをエージェント側に表示させるためである。

【0083】

D-3. カスタマー側・エージェント側ブラウザの同期

図19、20は、あるwebページが表示された状態で、カスタマーがリンクをクリック、URLを入力といった何らかの手段で他のページに移動した場合の同期処理手順を示すフローチャートである。なお、この例ではカスタマー側でページ移動が発生した場合を例に説明しているが、同期については、カスタマー、エージェントは対称であり、どちらからページの移動が発生しても処理手順は同じである。

【0084】

まず、カスタマーがページの移動を要求すると（ステップ511）、ブラウザ131がそのページをキャッシュ・マネージャー115経由で取得する。キャッシュ・マネージャー115は、UAIが存在するため、そのウェブページを所定のウェブサーバーから取得し、キャッシュ（キャッシュ・マネージャー115が管理している）に保管すると共に、アプレットPとスクリプト157を指定したタグを埋め込んで、カスタマー側のブラウザ131に返送する（ステップ513）。

【0085】

ブラウザは、新しいページを表示する前に表示されていたページを消去する。この際に、以前に表示されていたアプレットP136のstopメソッドが呼び出される。図5は、エージェント側端末170または、クライアント側情報端末13

0の同期時における構成要素をしめす機能ブロック図である。アプレットP193は、自分が停止することをツリー・マネージャー137に通知する（ステップ515）。ツリー・マネージャー137は、これによって表示されている内容に欠損が生じたことを検出し、相手側にロード開始状態を通知する（ステップ517～523）。これは、エージェント側にカスタマー側ブラウザの状態を表示するための処理であり、ページの同期を行うために必須のものではない。なお、アプレットP193とスクリプト191は、すでにキャッシュにあるため、実際には取得は発生しない（ステップ525）。

【0086】

移動先のページの表示が完了すると、ページのonloadまたはタイマーをトリガーとして、スクリプト191が起動される。スクリプト191は、部分木構造を取得する（ステップ527）。部分木構造とは、自分から、最上位のhtmlに到達するまでの親のURL、その親の子の数、その親の下での自分の位置をまとめたものである。図12、13にその例を示す。なお、この木構造に、最上位のWindowのName属性を含め、これを木の名前とすることによって、複数のブラウザウィンドウが存在する場合に、ウィンドウを判別して、同期させることも可能である。

【0087】

フレームが存在しない場合には、ブラウザ全体で1つしかhtmlが存在しないため、部分木構造と全体の構造は同一である。フレームが存在する場合には、末端のhtmlの数だけ、部分木構造が存在する。

【0088】

フレームをサポートしない場合には、ブラウザ全体で1つしかhtmlが存在しないため、アプレットにおいてロード元のURLを取得することができるため、スクリプト191の関与は不要である。スクリプト191は、この部分木構造を、アプレットP193に渡す。アプレットP193は、自分自身のハンドルと、部分木構造をツリー・マネージャー137に渡す（ステップ529）。

【0089】

ツリー・マネージャー137は、部分木構造を受け取る度に木構造を合成して行く（ステップ531）。図14、15にその過程の例を示す。末端のhtmlの数

だけの部分木構造を受け取ることによって、木構造が完成する。具体的には、スクリプト 1 3 6 は、自分とその親の情報だけを取得し、各htmlに挿入されたアプレット P に情報を渡す。アプレット P は、現在表示している h t m l の構造を管理するツリー・マネージャー 1 3 7 にその情報を引き渡す。なお、本発明の好適な実施例においてこのツリー・マネージャー（ツリー管理クラス） 1 3 7 はJava VM内でGUIを持たないスレッドとして実行され、各htmlのアプレットから共通に参照される。

【 0 0 9 0 】

図 1 2 に示すように、フレームがない場合の情報では、abc/def.htmから取得される部分木構造は、abc/def.htmとなり、これが生成されることとなる。図 1 3 のようにフレームがある場合の情報では、abc/index.htmには 2 つの子を持つが、図 1 4 のように 2 番目の子の URL は、スクリプト 1 9 1 にとって不明である（WindowのParent属性等、通常スクリプトがりようできる情報から親のURLの情報等を取得できる）。さらに、def/xyz.htmに埋め込まれたスクリプトからの情報がツリー管理クラスに渡されると、ツリーが完成する（図 1 5 ）。

【 0 0 9 1 】

木構造が完成し、同期中である場合には（ステップ 5 3 3 、 5 3 5 ）、ツリー・マネージャー 1 3 7 は新しい木構造をHTTPブリッジ 1 1 1 に送信する（ステップ 5 4 1 ）。HTTPブリッジ 1 1 1 は、図 1 1 のUAI-ソケット管理テーブル 2 7 8 から、UAIからセッション・マネージャー 1 1 7 へのソケット番号を取得してデータを送信する。これを受領したセッション・マネージャー 1 1 7 は、図 7 のセッション管理テーブル 2 1 0 から、セッションに参加している相手のソケットを特定して、データを転送する（ステップ 5 4 1 ）。

【 0 0 9 2 】

エージェント側のツリー・マネージャー 1 3 9 は、木構造を受信すると（ステップ 5 5 1 ）、現在の木構造と、受信した木構造を比較し（ステップ 5 5 3 ～ 5 5 7 ）、異なる場合には、ロード中状態を設定し（ステップ 5 6 1 ）、異なるhtml内の、あるいは異なるhtmlの子のhtml内のアプレット P のAppletContextを利用して、ページのロードを指示する（ステップ 5 6 1 ）。ただし、自分の側もロ

ード中である場合には、受信した木構造は記憶されるが、処理はされない（ステップ 5 5 3）。

【0 0 9 3】

この指示を受けてブラウザ 1 7 1 は、新しい Web ページをキャッシュ・マネージャー 1 1 5 経由で取得しようとする（ステップ 5 6 1）。キャッシュ・マネージャー 1 1 5 は、図 1 0 のキャッシュ管理テーブル 2 1 0 を参照して、すでにカスタマー側のブラウザ 1 3 1 で当該の html が取得済みであることを判定し、キャッシュに保管してあった html を取り出し、アプレット P とスクリプトを指定したタグを埋め込んでエージェント側のブラウザに返送する（ステップ 5 6 1）。なお、アプレット P とスクリプトは、すでにキャッシュにあるため、実際には取得は発生しない（ステップ 5 6 3）。

【0 0 9 4】

移動先のページの表示が完了すると、ページの onload または タイマー をトリガーとして、スクリプトが起動され、カスタマー側と同様にスクリプトが部分木構造を取得してアプレット P 経由でツリー・マネージャーに渡す（ステップ 5 6 9）。ツリー・マネージャー 1 3 7 では、自分の木構造が完成し、受信した木構造と同一になったかを判定する（ステップ 5 7 1、5 7 3）。同一になった場合には、木構造が揃った肯定応答（木構造の Ack）を HTTP ブリッジ 1 1 1、セッション・マネージャー 1 1 7 経由でカスタマー側に送信し、ロード中状態を終了する（ステップ 5 7 5）。なお、ロード開始側（カスタマー側）は、肯定応答を受信することで、ロード中状態が終了する。なお、すでに同期が開始されているには、ステップ 5 4 1 から処理を開始することで、相手側に木構造を送ることができる。

【0 0 9 5】

E. その他

以上、本発明をカスタマー・サポート・センタにおけるカスタマーとエージェントのコラボレーションに応用した例を用いて説明したが、本発明は、1 対多または多対多の電子会議等の同期にも応用可能な技術である。これは、セッション

管理テーブル、キャッシュ管理テーブルを、複数の参加者を管理できるように拡張し、セッション・マネージャーに、既存のセッションの一覧表作成機能、既存のセッションへの参加機能、既存のセッションからの脱退機能を追加する。そして、アプレットAで、既存セッションの一覧を表示して参加、脱退する指示ができるようにし、クライアントで、複数の相手からツリーを受信した場合に、どちらを優先するかを決定するロジックを追加するといった変更により実現することができる。

【0096】

【発明の効果】

以上説明したように、本発明によれば、クライアントシステムにコラボレーション・ソフトウェアを導入することなしにインターネット上でコラボレーションを行うことができる。

【0097】

【図面の簡単な説明】

【図1】 本発明の好適な実施例におけるコラボレーションシステムの概要を示す図である。

【図2】 本発明のコラボレーション・サーバーまたは情報端末のハードウェア構成の一実施例を示すブロック図である。

【図3】 エージェントがログオンするに際し動作する処理要素を含む本発明の好適な実施例におけるコラボレーション・システムのシステム構成を示す機能ブロック図である。

【図4】 エージェントとカスタマーがコラボレーションを開始した際に動作する処理要素を含む本発明の好適な実施例におけるコラボレーション・システムのシステム構成を示す機能ブロック図である。

【図5】 クライアント側情報端末の処理要素を示す機能ブロック図である。

【図6】 カスタマー側情報端末でURLの変更を伴うイベントが発生した際に動作する処理要素を含む本発明の好適な実施例におけるコラボレーション・システムのシステム構成を示す機能ブロック図である。

【図 7】 本発明の好適な実施例におけるセッション管理テーブルの概念図である。

【図 8】 本発明の好適な実施例におけるエージェント管理テーブルの概念図である。

【図 9】 本発明の好適な実施例におけるカスタマー待ち行列の概念図である。

【図 1 0】 本発明の好適な実施例におけるキャッシュ管理テーブルの概念図である。

【図 1 1】 本発明の好適な実施例における U A I - ソケット管理テーブルの概念図である。

【図 1 2】 本発明の好適な実施例における部分木構造を生成する過程を説明する概念図である。

【図 1 3】 本発明の好適な実施例における部分木構造を生成する過程を説明する概念図である。

【図 1 4】 本発明の好適な実施例における部分木構造を生成する過程を説明する概念図である。

【図 1 5】 本発明の好適な実施例における部分木構造を生成する過程を説明する概念図である。

【図 1 6】 本発明の好適な実施例におけるエージェントのログオンの手順を示すフローチャートである。

【図 1 7】 本発明の好適な実施例におけるカスタマーがログオンし、エージェントとコラボレーションの開始する際の動作手順を示すフローチャートである。

【図 1 8】 本発明の好適な実施例におけるカスタマーがログオンし、エージェントとコラボレーションの開始する際の動作手順を示すフローチャートである。

【図 1 9】 ある w e b ページが表示された状態で、カスタマーがリンクをクリック、URL を入力といった何らかの手段で他のページに移動した場合の本

発明の好適な実施例における同期処理手順を示すフローチャートである。

【図 2 0】 ある web ページが表示された状態で、カスタマーがリンクをクリック、URL を入力といった何らかの手段で他のページに移動した場合の本発明の好適な実施例における同期処理手順を示すフローチャートである。

【符号の説明】

- 1 1 0 コラボレーション・サーバー
- 1 1 1 HTTPブリッジ
- 1 1 3 セッション管理部
- 1 1 5 キャッシュ・マネージャー
- 1 1 7 セッション・マネージャー
- 1 1 9 U A I 生成部
- 1 3 0 カスタマー側情報端末
- 1 3 1 カスタマー側ウェブ・ブラウザ
- 1 3 3 カスタマー側制御ウインドウ
- 1 3 5 アプレット C
- 1 3 7 ツリー・マネージャー
- 1 3 8 アプレット A / C
- 1 3 9 通信部
- 1 4 0 ウェブ・サーバー
- 1 5 1、1 5 3 アプレット
- 1 5 5 コラボレーション・サーバー側ツリー・マネージャー
- 1 5 7 アプレット / スクリプト
- 1 6 0 コール・マネージャー
- 1 7 0 エージェント側情報端末
- 1 7 1 エージェント側ウェブ・ブラウザ
- 1 7 3 エージェント側制御ウインドウ
- 1 7 5 アプレット A
- 1 7 7 エージェント側ツリー・マネージャー
- 1 8 0 キュー・サーバー

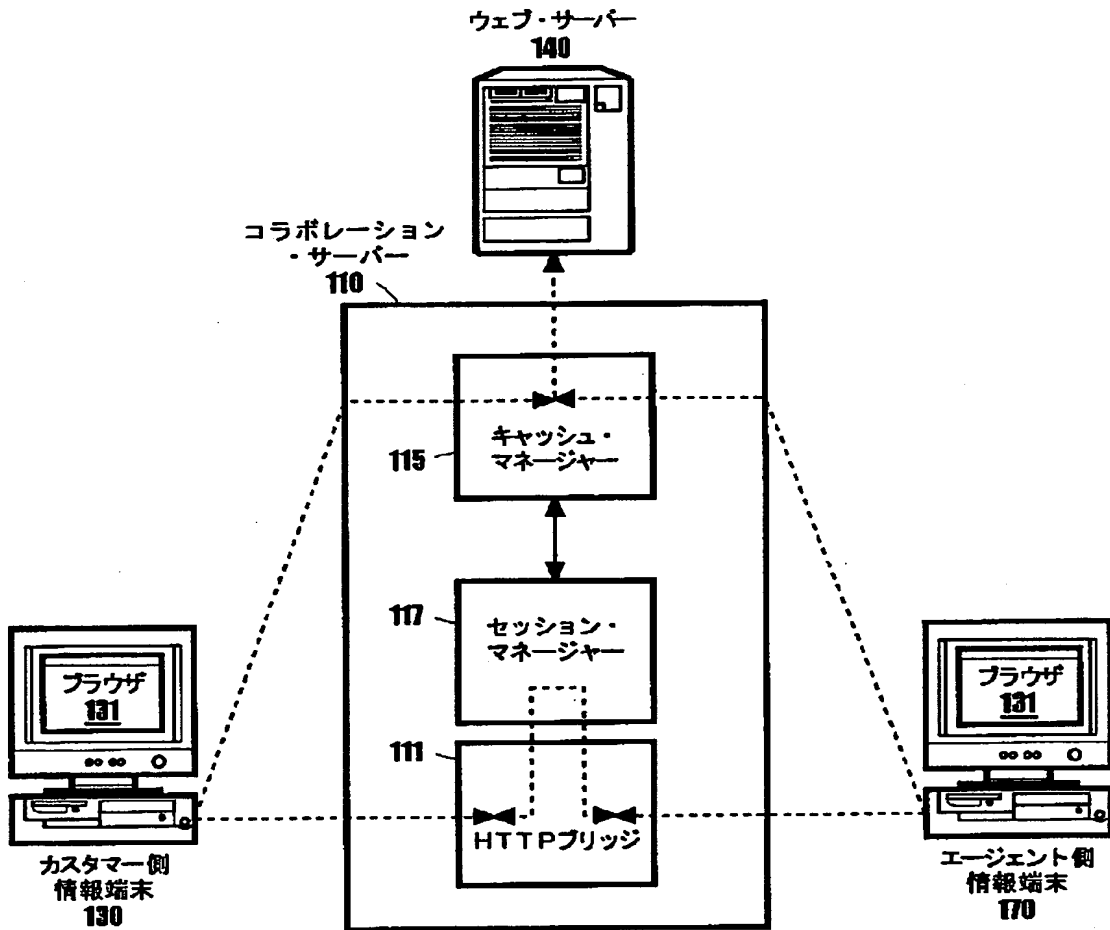
1 9 1 スクリプト

1 9 3 ページ代表アプレット P

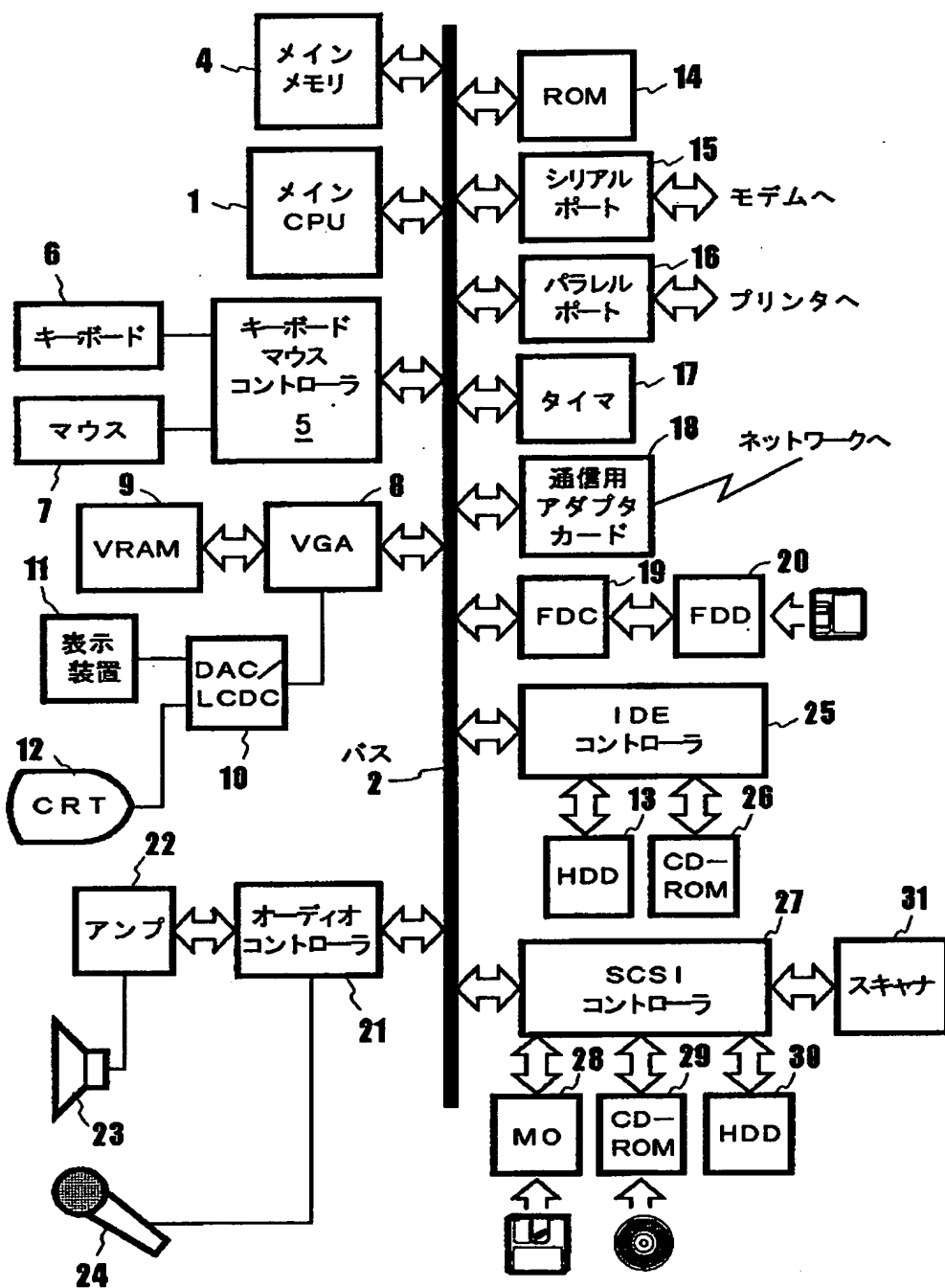
1 9 5 J a v a V M

【書類名】 図面

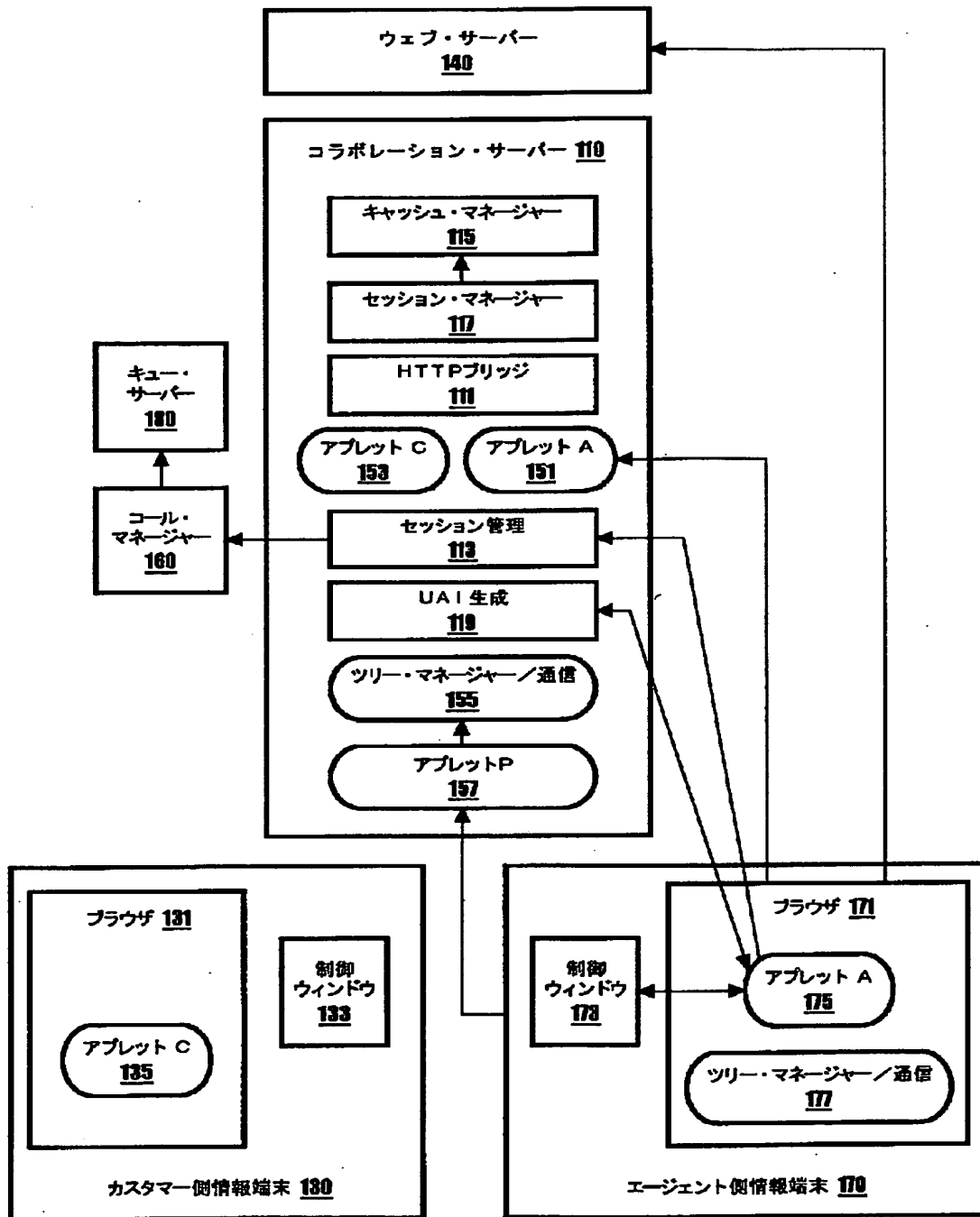
【図 1】



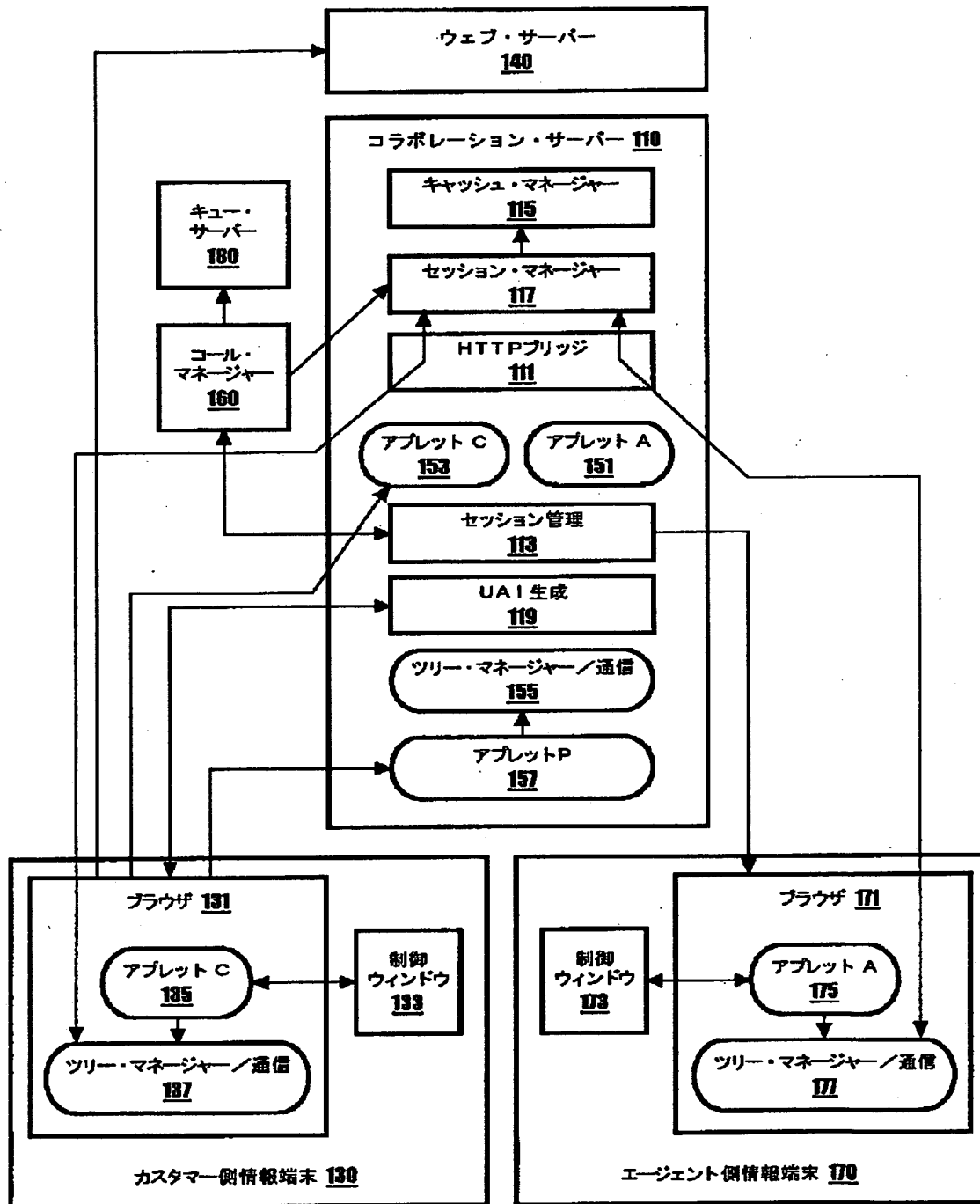
【図2】



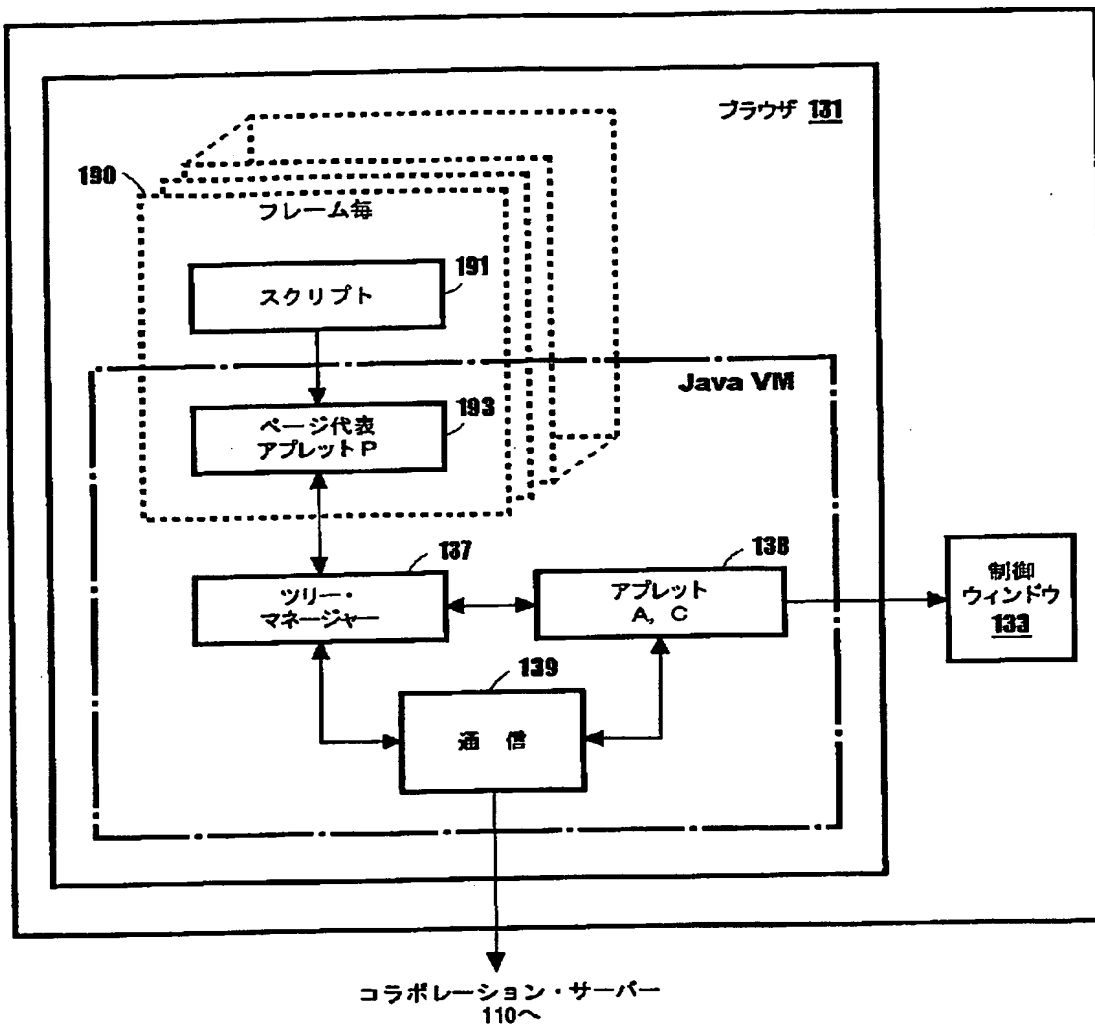
【図 3】



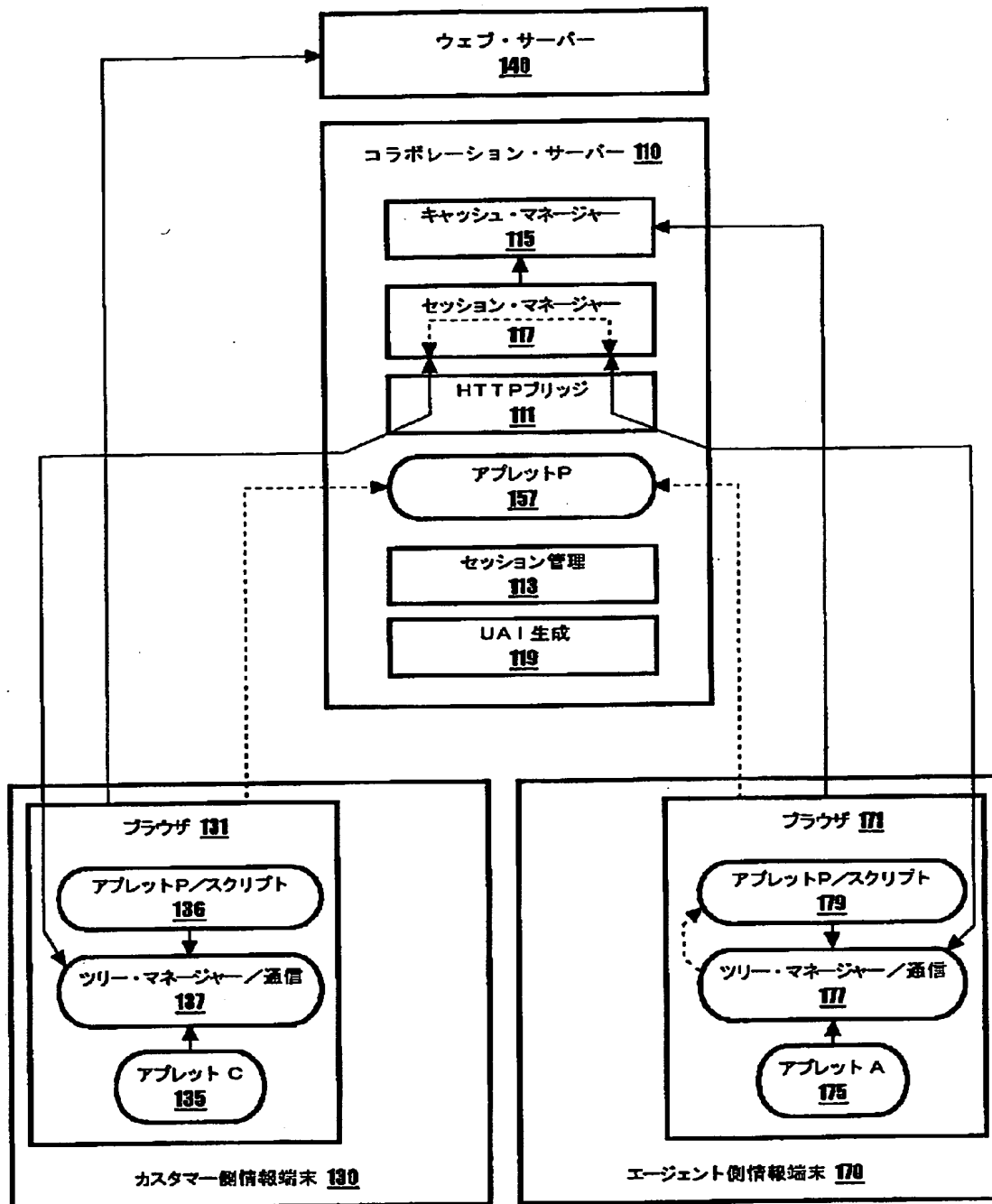
【図 4】



【図 5】



【図 6】



【図7】

セッション管理テーブル 210

セッションID 211	セッション状態 213	UAI1 215	ソケット1 217	UAI2 219	ソケット2 221
1	使用中	A123456789	00000011	000012345	00000012
2	空き				
3	空き				

【図8】

エージェント管理テーブル 230

エージェントID 231	待ち行列名 233	状態 235	UAI 237
AGENT01		ログオフ中	
AGENT02	PC	ログオン中	A123456789

【図9】

カスタマー待ち行列のエントリー 250

カスタマーID 251	待ち行列名 253	UAI 255
000000001	PC	000012345

【図10】

キャッシュ管理テーブル 270

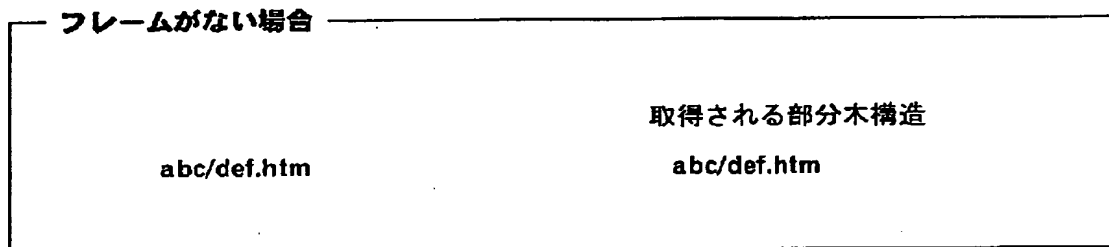
UAI1 271	UAI2 273
A123456789	000012345

【図 1 1】

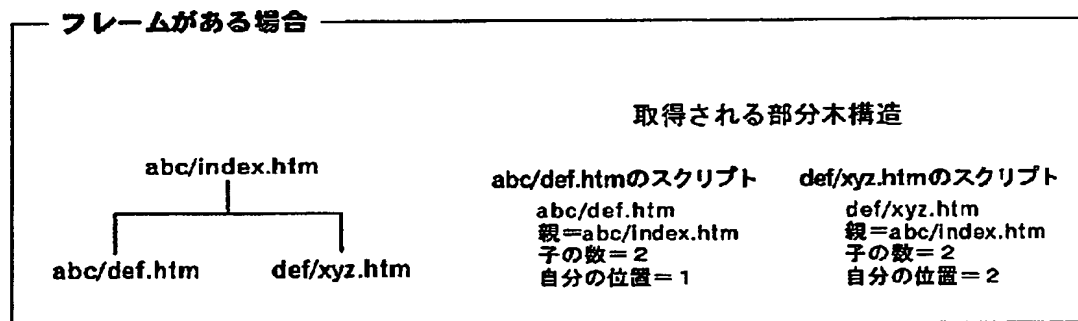
U A I - ソケット管理テーブル 280

U A I 281	ソケット 283
A123456789	00000022

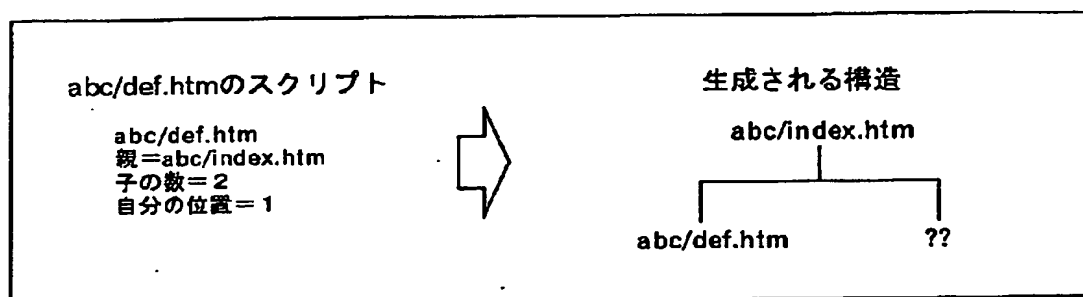
【図 1 2】



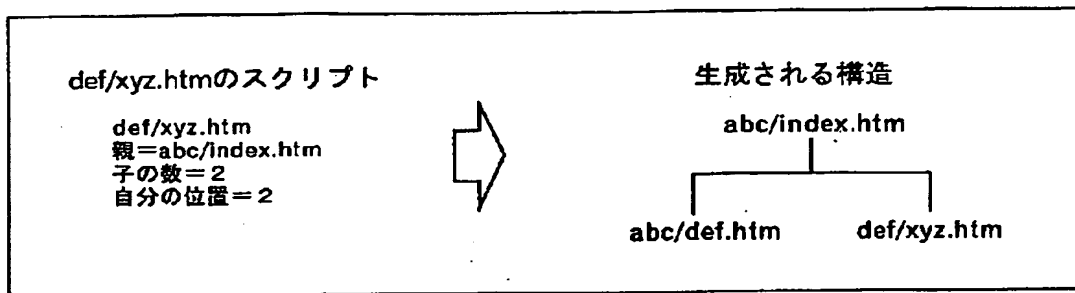
【図 1 3】



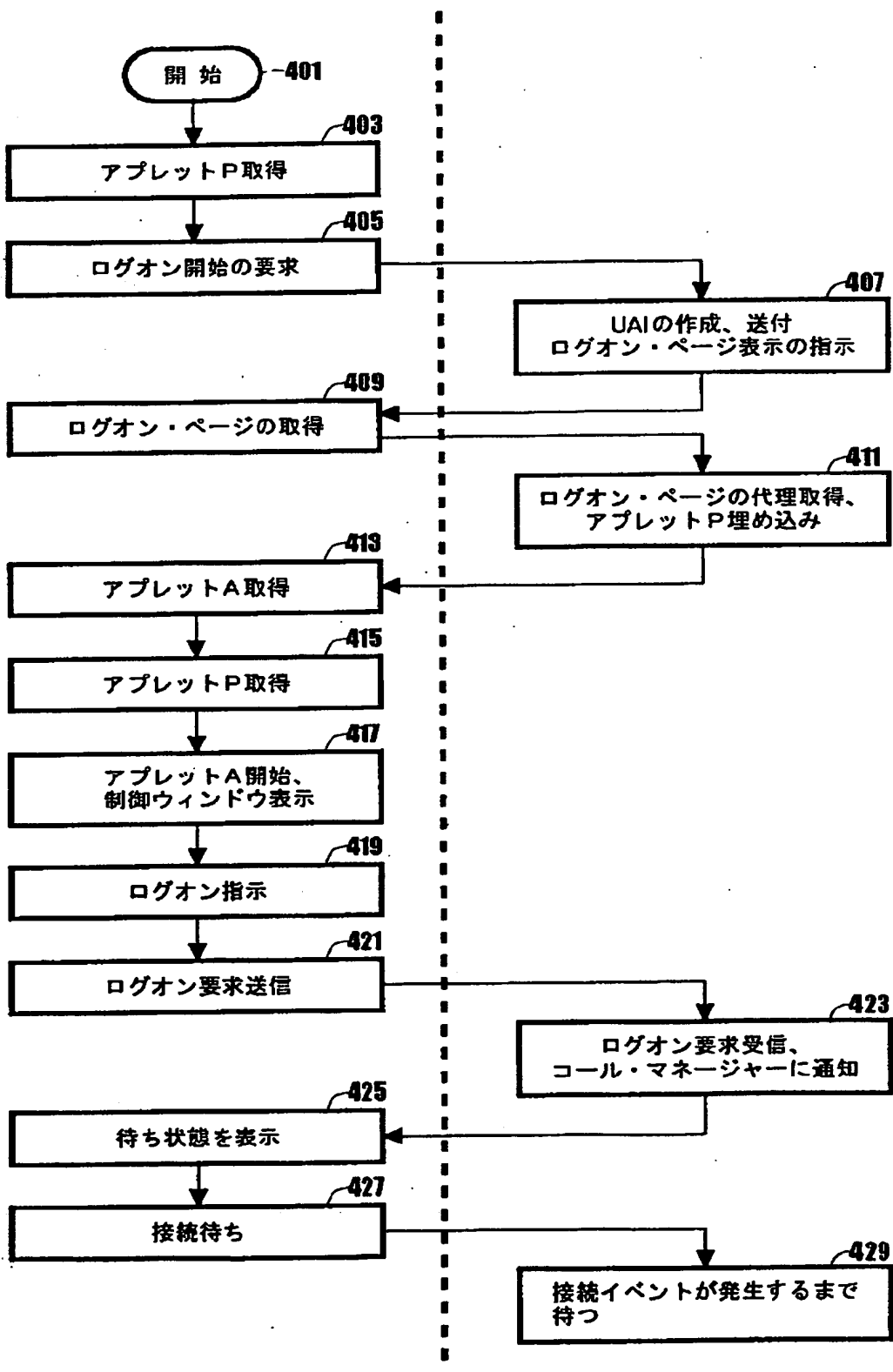
【図 1 4】



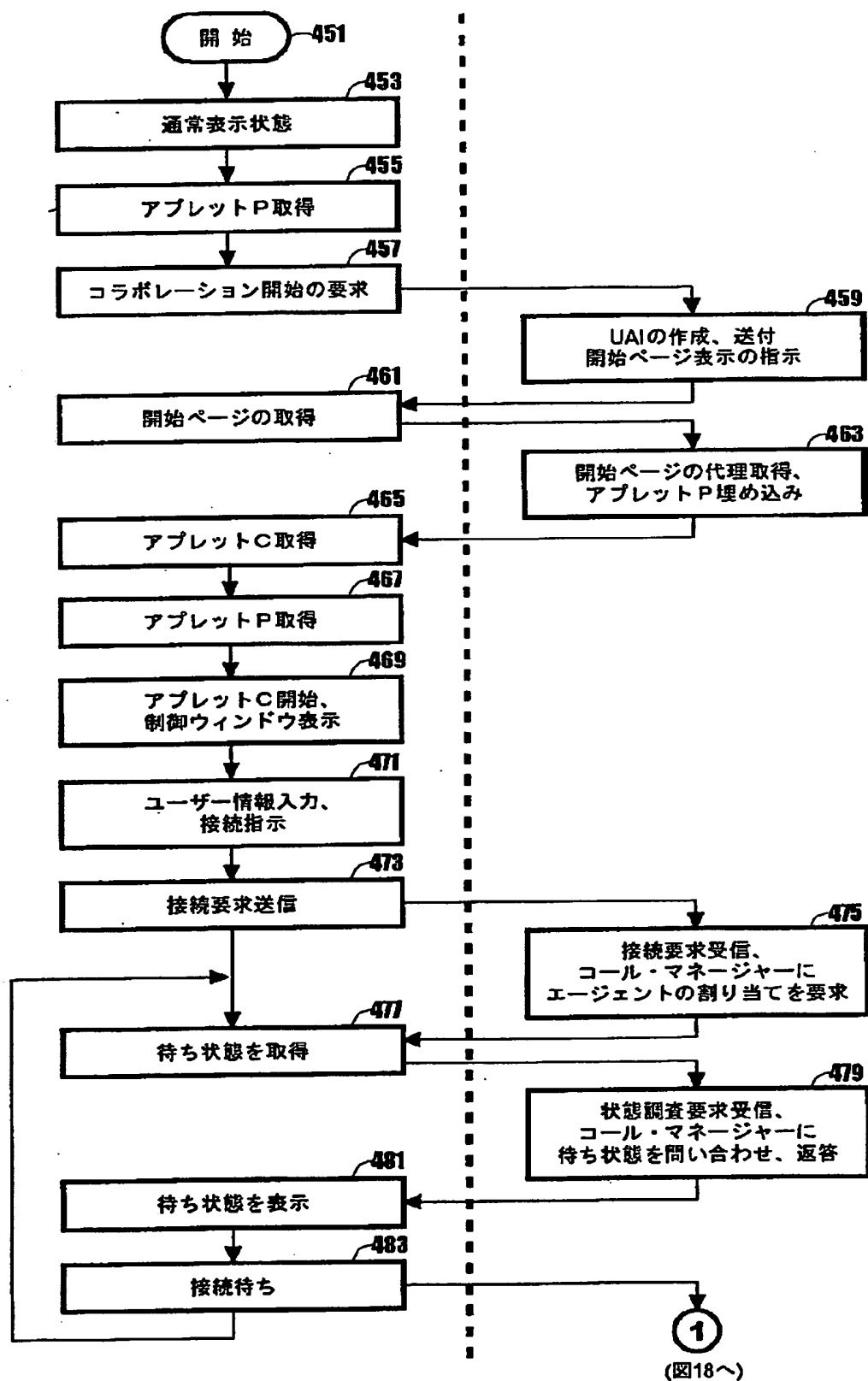
【図 1 5】



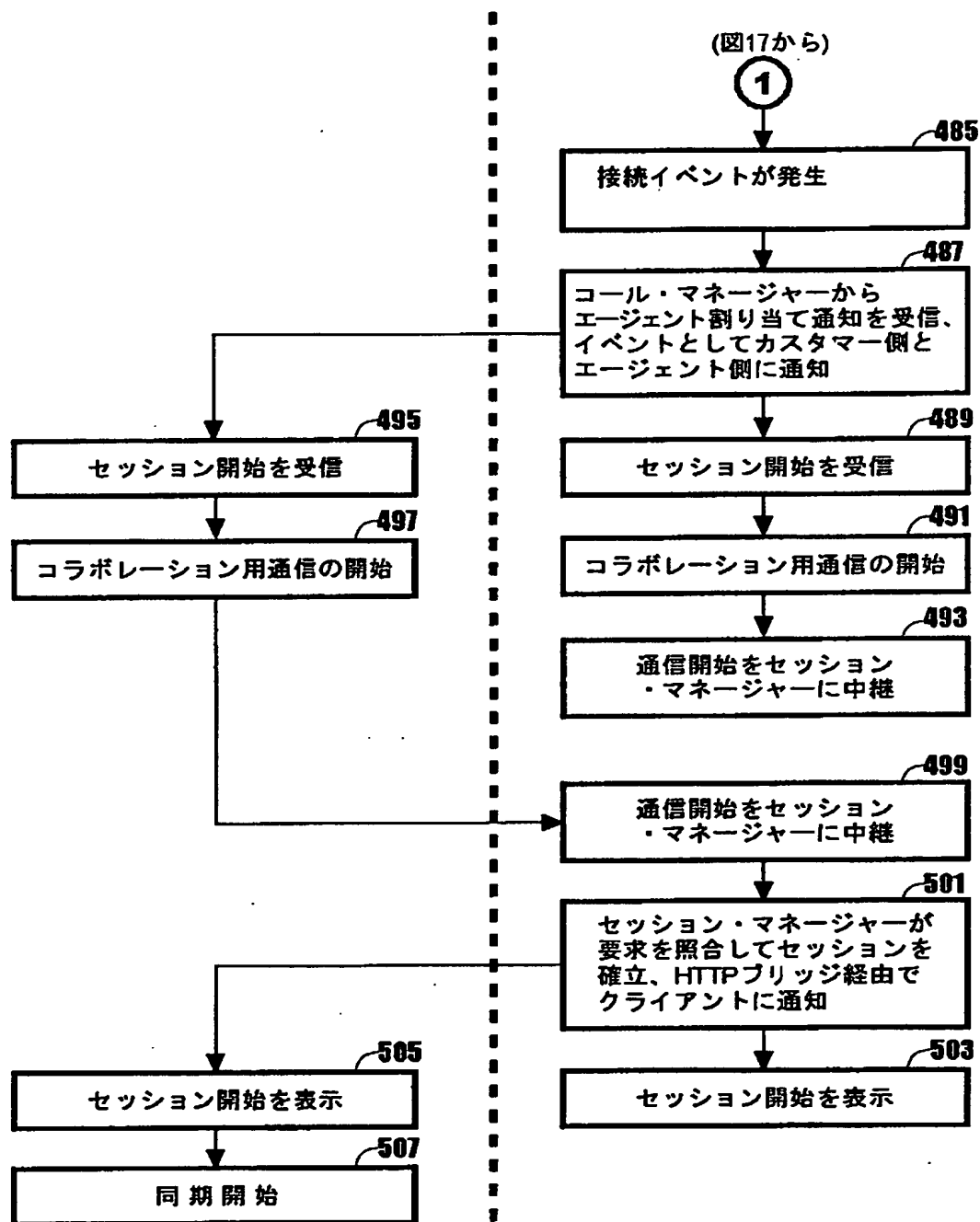
【図16】



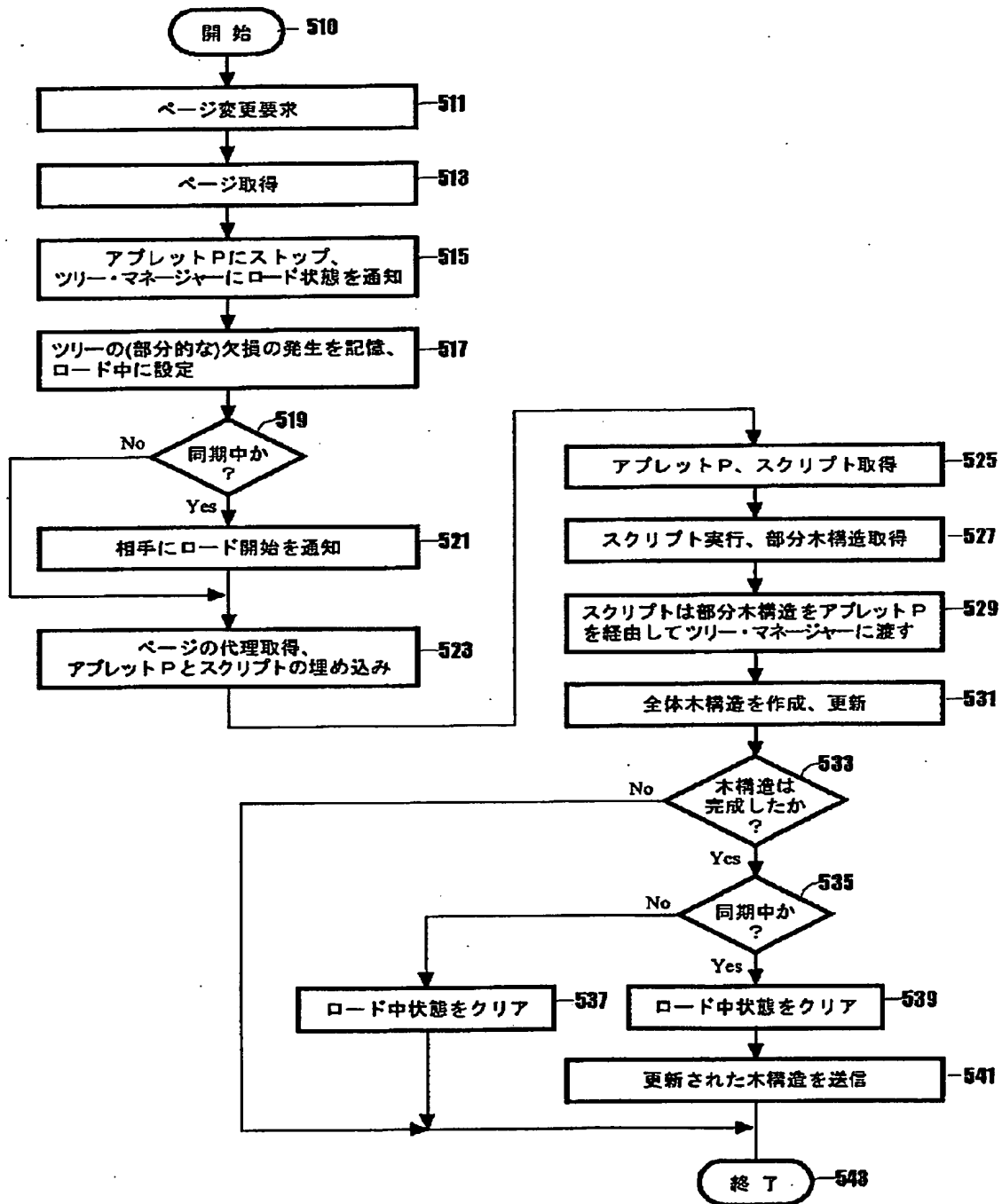
【図 17】



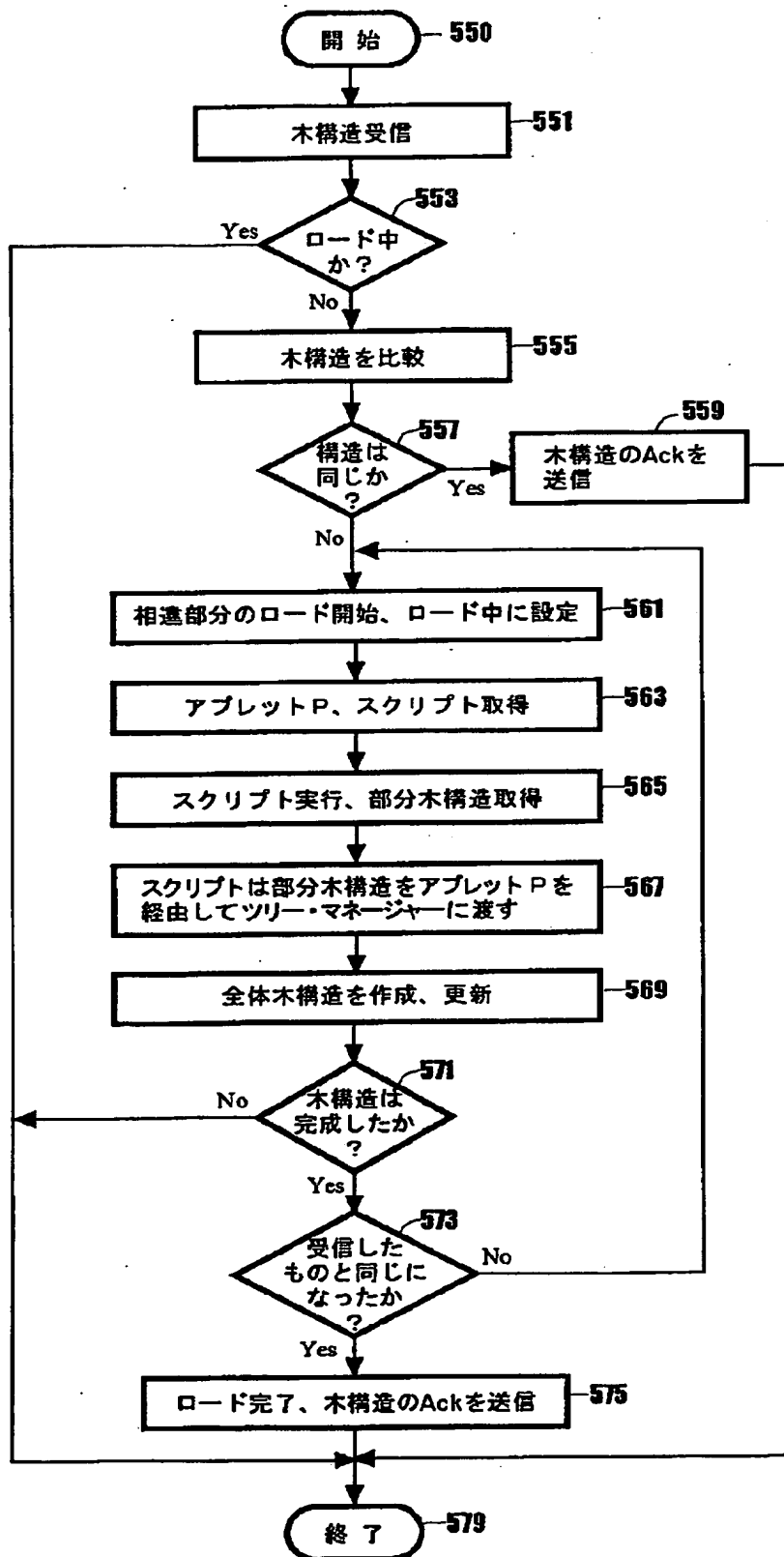
【図 18】



【図19】



【図20】



【書類名】 要約書

【要約】

【課題】 予め情報端末に特別なソフトウェアを導入することなしに、遠隔地にある複数の情報端末がコラボレーションを行なうことを可能にする。

【解決手段】

カスタマー側ブラウザ 1 3 1 とエージェント側ブラウザ 1 7 1 は、コラボレーション・サーバー 1 1 0 経由で接続され、ウェブ・サーバー 1 4 0 もコラボレーション・サーバー経由でアクセスするように制御されている。コラボレーション・サーバーは、ブラウザ 1 3 1、1 7 1 があるページを参照するに際し、ページの変化を検出するアプレットと、コラボレーションを行なう相手側の変更後のページ情報に対応したページを表示させるクライアント・コントローラーを埋めこむ。カスタマー側またはエージェント側ブラウザにおいて、ページの変化が検出されると、このアプレットが起動され、コラボレーション・サーバー経由で変更後のページ情報を相手側に送られ、相手側のクライアント・コントローラーは、変更後のページを表示するようにブラウザを制御する。

【選択図】 図 1

特2000-036314

認定・付加情報

特許出願の番号	特願2000-036314
受付番号	50000166222
書類名	特許願
担当官	第七担当上席 0096
作成日	平成12年 2月16日

<認定情報・付加情報>

【提出日】	平成12年 2月15日
-------	-------------

次頁無

出 願 人 履 歴 情 報

識別番号 [390009531]

1. 変更年月日 1990年10月24日

[変更理由] 新規登録

住 所 アメリカ合衆国10504、ニューヨーク州 アーモンク (番地なし)

氏 名 インターナショナル・ビジネス・マシーンズ・コーポレイション